

Факултет инжењерских наука Универзитета у Крагујевцу

Вељко Д. Димовић

**Примена природних корисничких интерфејса у
развоју пословних апликација**

мастер рад

Крагујевац, 2013.

Факултет инжењерских наука Универзитета у Крагујевцу



Назив студијског програма: Машинско инжењерство

Ниво студија: Мастер академске студије

Модул: Информатика у инжењерству

Предмет: Информациони системи

Број индекса: 321/2010

Вељко Д. Димовић

Примена природних корисничких интерфејса у развоју пословних апликација

мастер рад

Комисија за преглед и одбрану:

1. Др. Милан Д. Ерић

2. _____

3. _____

Датум одбране: _____

Оцена: _____

У оквиру овог мастер рада кандидат треба да:

Проучи и презентира развој пословних апликација применом природних корисничких интерфејса (Natural User Interface – NUI). Рад треба да обухвати приказ природних корисничких интерфејса, предности и недостатке истих, неопходне технологије за њихово остваривање као и начин примене. На конкретном пословном информационом систему приказати примену NUI-а.

Препоручена литература:

1. Wigdor D., Wixon D.: Designing Natural User Interfaces for Touch and Gesture, Elsevier Inc., Burlington, 2011.
2. Keen S., Hall J., Perry P.: An Introduction to Programming Natural User Interfaces, Springer, New York, 2011.
3. Лазаревић Б.: Базе података, ФОН Београд, Београд 2003.
4. Полишчук Ј.: Пројектовање информационих система, Електротехнички факултет, Подгорица, 2007.

Крагујевац, децембар 2012.

ментор:

Др Милан Д. Ерић

Резиме

У овом раду је анализиран начин рада природних корисничких интерфејса (Natural User Interface – NUI) са посебним освртом на специфичности њихове примене у пословним апликацијама. Истраживање се састојало од анализе најпопуларнијих комерцијално доступних сензора најновије генерације и анализе алгоритамских стратегија за њихову примену. У раду је представљено на који начин је потребно приступити дизајнирању апликација које користе NUI технологије и шта је потребно променити у основним концептима и комуникацији са корисником преко корисничког интерфејса. Резултати дате анализе су примењени кроз комплетан пример развоја и употребе пословне апликације која користи NUI. Апликација показује начин рада и употребе интерактивног излога који користи Microsoft Kinect и његову примену у маркетиншке сврхе.

Кључне речи: Природни кориснички интерфејс (Natural User Interface – NUI), пословне апликације, кориснички интерфејс, Microsoft Kinect, интерактиван излог, маркетинг.

Abstract

This study analyses natural user interfaces (NUI) with special focus on the use of NUI in business applications. The research included an analysis of the most popular state-of-the-art sensors which are commercially available as well as an analysis of the algorithmic strategies for their use in applications. The study presents the methodology that should be applied when designing applications which utilize NUI technologies as well as what should be changed in basic concepts and communication with the user through the user interface. The results of this analysis were applied in the project which illustrates the development and use of a business application which utilizes NUI. This application demonstrates the operation and use of an interactive storefront and how it can be utilized for marketing purposes with Microsoft Kinect.

Key words: Natural User Interfaces (NUI), business applications , user interface , Microsoft Kinect, interactive storefront, marketing .

Изјављујем да сам овај мастер рад урадио самостално, служећи се стеченим знањем и искуством као и наведеном литературом.

321/2010 Вељко Димовић

Садржај

1 УВОД.....	1
2 ПРИРОДНИ КОРИСНИЧКИ ИНТЕРФЕЈС - NUI.....	3
2.1 ПРЕДНОСТИ И НЕДОСТАЦИ ПРИРОДНИХ КОРИСНИЧКИХ ИНТЕРФЕЈСА	6
2.2 СЕНЗОРИ ЗА NUI	7
2.2.1 Microsoft Kinect	8
2.2.2 Softkinect.....	11
2.3 КОРИСНИЧКИ ИНТЕРФЕЈС ОРЈЕНТИСАН КА NUI.....	12
2.3.1 Постојаност објекта	12
2.3.2 Промене у систему управљања фајловима.....	13
2.3.3 Интерфејс орјентисан ка садржају (Content-Centric UI)	14
2.3.4 Коришћење основних вештина	17
2.3.5 Учење нових вештина.....	18
2.3.6 Задаци за учење употребе интерфејса	21
3 АПЛИКАЦИЈА: ИНТЕРАКТИВНИ ИЗЛОГ.....	25
3.1 ПОСТОЈЕЋИ ПРОБЛЕМИ РЕКЛАМИРАЊА У ИЗЛОЗИМА И РЕШЕЊЕ ПРОБЛЕМА	25
3.1.1 Решење проблема	26
3.2 КОНФИГУРАЦИЈА ИС.....	27
3.3 АГЕНТ АПЛИКАЦИЈА	28
3.3.1 Слој за приступ подацима	29
3.3.2 Статистички модули и приказивања информација	30
3.4 ИЗЛОГ.....	32
3.4.1 AIDA.....	37
3.4.2 Иницијализација Kinect сензора и употреба у апликацији	39
3.4.3 Управљање покретима.....	40
3.4.4 Гејмификација.....	45
3.4.5 Цртање објеката у надграђеној стварности	46
3.4.6 Праћење подацима (Data tracking)	48
3.5 ЦЕНТРАЛНИ СЕРВЕР.....	50
3.5.1 База података	50
3.5.2 Сервис за комуникацију са базом података.....	55
3.6 ПРОБЛЕМИ ПРИЛИКОМ РЕАЛИЗАЦИЈЕ И ЕКСПЛОАТАЦИЈЕ	57
4 НАЧИН И УПОТРЕБА	59
4.1 ИНСТАЛАЦИЈА ИНТЕРАКТИВНОГ ИЗЛОГА.....	59
4.1.1 Конфигурација мрежних параметара и иницијални тестови	63
4.2 УПУТСТВО ЗА УПОТРЕБУ FRONTPANEL АПЛИКАЦИЈЕ.....	64
4.3 ОДРЖАВАЊЕ ЦЕНТРАЛНОГ СЕРВЕРА.....	68
5 БУДУЋНОСТ ПРИМЕНЕ И РАЗВОЈА	70
5.1 ХАРДВЕР БУДУЋНОСТИ	70

5.2 ПРАВАЦ РАЗВОЈА СОФТВЕРА	71
6 ЗАКЉУЧАК.....	73
7 ДОДАТАК А - НАДГРАЂЕНА СТВАРНОСТ (AR - AUGMENTED REALITY).....	74
7.1 ПРИМЕНА НАДГРАЂЕНЕ СТВАРНОСТИ У РАЗЛИЧИТИМ ОБЛАСТИМА.....	75
8 ЛИТЕРАТУРА	77

1 Увод

Развој информационих система у 21-ом веку прати драстичан и изузетно убрзан развој комплетне индустрије. Може се заправо рећи да је развој других грана индустрије у великој мери сразмеран развоју информационих технологија, такође се може рећи да велика достигнућа су последица захтева диктираних из индустрија забаве, образовања и наравно војне индустрије.

У овом раду направљена је паралела између најновијих информационих технологија и пословних апликација. Све ово је спојено у једну функционалну целину која се састоји са једне стране од природних корисничких интерфејса се сензорима које нуди последња генерација играчких конзола, припојена је логика функционисања информационих систем који представљају пословну логику и намену апликације. И све ово је искоришћено у сврху маркетинга за креирање једног комплетно новог производа који треба да промени начин функционисања одређених тржишта.

Да би све ово било могуће потребно је познавање следећих области:

- Архитектуре информационих система и пословних апликација.
- Креирање природних корисничких интерфејса.
- Функционисање и програмирање сензора за управљање природним корисничким интерфејсима.
- Технологије пројекције велике резолуције.
- Надграђена стварност (Додатак А - поглавље 6).
- Маркетиншке стратегије и гејмификација кампања.

Све битне компоненте овог рада су покривене у теоријским основама у прва два поглавља, док остатак рада се бави практичном реализацијом пројекта. Да би смо могли да разматрамо процес креирања информационих система морамо кренути од основа шта су заправо информациони системи.

Информациони систем (ИС) - Представља интеграцију информационих технологија, пословних процеса и људства које имају за циљ да изађу у сусрет и задовоље потребе предузећа или компанија у њиховом пословању.[1]

Постоје различити типови информационих система, на пример:

- Системи подршке одлучивању
- Канцеларијски системи
- Системи управљања знањем
- Системи за обраду трансакција
- Системи за управљање базама података
- Експертски системи
- Географски информациони системи
- Претраживачи интернета итд.

Заједничко за све ове системе су информационе технологије које имају за циљ да помогну човеку у обављању одређених операција а немају за циљ да замене човека у одређеном процесу. Ово се постиже обављањем операција које човеку треба више времена да изведе или није у стању да их изведе. Заправо да ИС иду у корист процесу запошљавања говори то да су многе компаније

знајући колико битне могу бити квалитетне информације отворила нова радна места намењена оператерима информационих система.

Процес развоја информационих система је интегралан процес који у себе укључује велики број дисциплина чији избор зависи од намене самог система.

Услед великог број нових изума или примена старих у комплетно нове сврхе дошло је отварања посебне гране информационих технологија. Пре мање од једне деценије нису се могли замислити мобилни телефони процесорске снаге јаче него професионални стони рачунари са краја претходне деценије. Екрани огромних резолуција, играчке конзоле које прате покрете или препознају све гласовне команде. Доступне су за цену просечног РС рачунара. Све ово спојено у јединствен информациони систем који за основну намену нема играње игрица делује поприлично далеко. Али са почетком ове деценије било је могуће направити искорак ка новом начину управљања информационим системима, природни кориснички интерфејси су постали стварност! (Natural User Interface - NUI.) Шта се подразумева под називом природни кориснички интерфејс?

2 ПРИРОДНИ КОРИСНИЧКИ ИНТЕРФЕЈС - NUI

Први сусрет са концептима NUI се дешава још 80-их година двадесетог века када је Стив Ман (Steve Mann) креирао неколико стратегија корисничког интерфејса користећи природну интеракцију као алтернативу за интерфејс командне линије CLI(Command Line Interface) или графички кориснички интерфејс GUI(Graphical User Interface).[2]

Ман је назвао свој рад као "Природни кориснички интерфејс" или "Директни кориснички интерфејс". Манова технологија користи реч Природно као референцу на истовремено природне људске покрете и природу која окружује човека.



Слика 2-1Еволуција корисничких интерфејса

У 2006 години Кристијан Мур (Christian Moore) је покренуо отворену истраживачку заједницу да би проширио дискусију и развој повезан са NUI технологијама. 2008 године Microsoft је представио прве комерцијалне уређаје са NUI технологијама и најавио да ће ово бити следећи еволуциони корак у развоју као што је био прелазак са CLI на GUI . Наравно ово је благо преувеличано јер није могуће направити NUI без графичких ефеката и визуелних елемената.

У CLI корисник је морао да научи вештачки начин уноса и интеракције, тастатура и серије кодификованих улазних команди имала је ограничене могућности док је синтакса била веома строга.

Затим са појавом миша омогућен је GUI, корисницима је било знатно природније да науче кретање мишем и доживе бољу интеракцију са интерфејсом. Овај вид интеракције је захтевао контакт на екрану између садржаја и објекта.

Права природна интеракција са рачунаром у догледно време сигурно није могућа али употреба термина NUI је постала заједничка за сваку интеракцију електронских уређаја које имају могућности, Multi-Touch екрана, препознавања гестова корисника или гласовних наредби односно уређаји који не користе класичан GUI и не захтевају поклапање објеката на радној површини се могу сматрати пионирским уређајима који користе NUI технологију.

Multi-Touch технологија представља већ више деценијски доступну технологију која има дугу историју али због концепта тадашњих рачунара овакав приступ је сматран превише

компликованим у поређену са класичним "single touch" интерфејсима и може се сматрати самим почетком примене ових технологија.

У биће наведени примери интерфејса који се могу сматрати NUI технологијом:

Perceptive pixels - Ова технологија је демонстрирана 2006 године и базирана је на Multi-Touch интерфејсима, приказан је одређени број интеракција са садржајем на екрану користећи истовремено и директан контакт као и гестове. Ова технологија је омогућавала манипулацију објеката симулацијом гестова прстима за смањење и повећање садржаја превлачећи прстима у жељеном смеру. Циљ ове технологије је био да дочара да овај начин комуникације може бити много више интуитиван него до сад и да постојећи уређаји могу бити замењени екранима који препознају ову интеракцију.



Слика 2-2 *Perceptive pixels* технологија

Microsoft PixelSense - Ова технологија је узела сличне основне идеје интеракције са садржајем само је била у могућности да препозна уређаје спуштене на површину. На овај начин корисници су могли да имају исту интеракцију са површином као у претходном примеру уз додатак да ново лоцирани уређаји постају предмет интеракције и контролног механизма.

На пример ако би сте спустили винску чашу на сто компјутер је препознаје и приказује садржај повезан са објектом. Такође постављање објекта на површину изазива реакцију "мапе" приказивања интерфејса тако да садржај прати и одговара активностима на столу. Ово се може сматрати уводом у NUI технологије.

Ова технологија поред вишеструког додира омогућава и различити број истовремених корисника оптимизујући садржај и приказујући га према кориснику.

Наравно садржај оваквог типа је зависио од базе објеката које је било могуће препознати али интеракција корисника без обзира на положај је била универзална.



Слика 2-3 Microsoft PixelSense

3D Immersive Touch - је дефинисан као директна манипулација 3D виртуалним окружењем користећи single или multi-touch површине као hardware у овом окружењу. Ову технологију можемо сматрати претечом управљања данашњих уређаја са екранима осетљивим на додир и играњем изузетно напредних игрица на њима. Такође први пут се уводи и концепт 3D корисничког природног интерфејса.

Xbox Kinect - је производ за Xbox играчку конзолу која користи специјалне гестове за интеракцију са игрицом уместо контролера. Заправо произвођач тврди да за овај нови револуционарни начин играња није потребан контролер. Такође Kinect поседује и низ микрофона који омогућавају гласовну контролу. О овом уређају биће знато више речи накнадно.



Слика 2-4 Kinect игрице

2.1 Предности и недостаци природних корисничких интерфејса

Револуција природних корисничких интерфејса је неизбежна. Сведоци смо историје да је свет мигрирао са текстуалних на графичке апликације зато што су биле способније и лакше за свакодневну употребу на свакодневном нивоу. Са чисто пословне стране, уколико имате текстуалну апликацију а ваша конкуренција има графичку коју сви желе, тржиште ће од вас захтевати да се прилагодите и урадите исто. Иста ствар се сада дешава поново, само што сад GUI је застарела технологија а NUI представља способнију, лакшу за учење и једноставнију технологију.

Транзиција корисничких интерфејса је област веома погодна за истраживање и кроз литературу се може наћи много ставки које су за и против различитих стилова интерфејса. У наставку је описана листа позитивних ствари код директне манипулације која је један од основних карактеристика NUI-а, посебно екрана осетљивих на додир. О начинима креирања интерфејса ће мало касније бити речи, погледајмо предности сад:

- Нови корисници могу научити основне функционалности веома брзо, најчешће кроз демонстрацију искуснијег корисника.
- Експерти могу да раде веома брзо и да извршавају широк дијапазон задатака, чак дефинишући нове функције и могућности.
- Познаваоци и повремени корисници могу задржати познате концепте.
- Порукe о грешкама су ретко потребне.
- Корисници одмах могу видети да ли њихове акције воде ка циљу, и уколико не, могу једноставно да промене смер својих активности.
- Корисници смањују нелагодност приликом коришћења јер је систем тако разумљив и акције су лако повратне. [3]

Ово је сјајна листа атрибута природних корисничких интерфејса, осим једне ствари: цитирани рад је написан 1982 године и написао га је Бен Шнајдерман а он је описивао предности директне манипулације у надлазећој револуцији графичких корисничких интерфејса.

Када се погледа из ове перспективе, GUI је директнији од CLI, али NUI може бити много директнији од GUI-а. Данас ми не мислимо о GUI-ју као веома директном, практично зато што смо научени на његове стандарде, начине интеракције и нарочито због не оптималности интеракцијских шаблона које дизајнери апликација нису испоштовали и усвојили у смеру праве директне манипулације.

На пример, идеја да GUI мора ретко да приказује грешке је тотално погрешна, NUI има предност над GUI у извршавању задатака опште намене. Нове технологије улаза чине овај вид флексибилнијим и компатибилнијим него класични миш и тастатура.

Фокус на природно понашање чини интерфејс лакшим за учење и свакодневни задаци су једноставнији за обављање.

GUI и архитектура орјентисана на миша имаће и даље улогу у будућности, само што ће та улога бити ограничена и специјализованија него данас. Уколико постоји двоумљење око креирања апликације са GUI или NUI корисничким интерфејсом ево неких да или не питања које треба поставити да би се ова недоумица решила:

- Да ли ће апликација бити коришћена само на класичном компјутеру са мишем и тастатуром?
- Да ли апликација мора да буде дизајнирана око управљања мишем због било ког одређеног разлога осим традиције?
- Да ли ће апликацију користити корисници који су добро тренирани?
- Да ли је заоставштина Виндоуз дијалога, менија и иконица заиста толико битнија него једноставност коришћења апликације.
- Да ли брзина преласка корисника са старих апликација на нову битнија од корисничког искуства.

Уколико је било који одговор од ових НЕ, онда би апликација имала већу корист од природних корисничких интерфејса.

У неким случајевима сигурно је да миш или сличне технологије показивача би биле неопходније због прецизног показивања, али остатак апликације би могао да буде природан кориснички интерфејс.

На крају овај део би могли да закључимо следећим констатацијама, NUI је нови начин размишљања како вршимо интеракцију са садржајем, они имају за циљ да искористе постојеће вештине за правилну интеракцију са садржајем.

NUI је стил интеракције и нема никакве везе са било којим улазним уређајем.

Традиционални кориснички интерфејс ће и даље бити ту само ће се користити за специјализоване задатке а NUI ће преузети примат у великом броју општих корисничких апликација.

2.2 Сензори за NUI

Као што је претходно описано постоје четири врсте технологија које могу користити за NUI. У поглављу - Кориснички интерфејс оријентисан ка NUI (2.3) детаљније је говорено о начину креирања корисничког интерфејса за ову врсту апликација, такође споменути примери се односе на примену технологија додир, нарочито на **Microsoft PixelSense** и **3D Immersive Touch**. Сензори који су доступни у нашој земљи за ове технологије имају још увек високу цену а оне најквалитетније није могуће и набавити. Овде наравно не причамо о развоју апликација за таблет уређаје и паметне телефоне који колико год да су напредни и у одређеној мери користе NUI нису наша примарна разматрана тема. Разлог овога је што желимо да покажемо могућности примене и утицај NUI на пословне апликације и системе.

Због овога одлучено је да се демонстрира технологија праћена покрета и гласа. Разлога за ово има више:

- Технологије праћења покрета никада у својој историји нису били јефтиније него сад, где се камере неопходне за ове операције крећу у распону од 20000 - 50000 динара.
- Већа могућност иновације у процесу креирања интерфејса и саме апликације. Ова технологија је била некад примењивана само у скупим холивудским филмовима, данас је корисничка електроника нижег реда.
- Могућност манипулације једноставним покретима, сложеним гестовима и гласом.
- Довољно развијени SDK пакети да несметано омогућавају развој апликације и омогуће праву пропорцију између утрошеног времена и квалитета добијене апликације.

- Прави тржишни моменат. Ова технологија се користи само у индустрији игрица и то од краја 2011 године. Док како је свима познато апликације за екране осетљиве на додир постоје већ велики број година. Сад наравно у којој оне мери имплементирају NUI то је посебна тема.
- Велики практични пример демонстриран у поглављима **Error! Reference source not found.** и 4 јер рађен у овој технологији.

2.2.1 Microsoft Kinect

Kinect је улазни уређај који омогућава праћење покрета који је креирао Microsoft за свој Xbox 360 и персоналне рачунаре. Креиран по углед на изглед интернет камера и као периферни уређај за Xbox 360, омогућава корисницима да контролиши и врше интеракцију са конзолом без потребе да користе контролере и то искључиво користећи NUI и технологију препознавања гласа и гестова. Пројекат је имао за циљ да прошири циљну групу корисника Xbox 360, Kinect се такмичи са уређајима за праћења покрета које су пласирали PlayStation и Wii. Такође пре годину дана Microsoft је објавио да ће бити верзија Kinect уређаја и за кућне рачунаре.

Kinect је оборио рекорд тако што је постао најбрже продавани уређај потрошачке електронике у историји са укупном продајом од 8 милиона за два месеца.[5]

Microsoft је објавио SDK за Windows 7 јуна 2012 године, овај SDK омогућује писање апликација у C++/CLI, C#, или Visual Basic .NET-у.



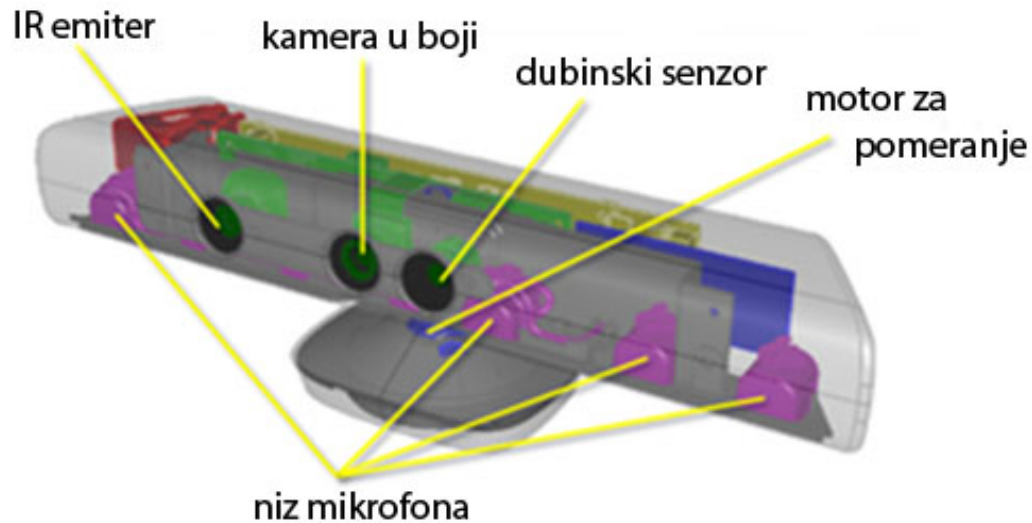
Слика 2-5 Изглед Microsoft Kinect уређаја за Xbox 360

2.2.1.1 Технологија

Kinect се надограђује на софтверски технологију креирану за интерну употребу од стране Microsoft Game Studios и технологије даљинских камера развијених у израелској компанији PrimeSense. Они су развили систем који може да пресретне и препозна одређене гестове, контролише уређаје без употребе било ког контролера у рукама тако што користи инфрацрвени пројектор, камеру и специјални чип који омогућава праћење објеката и индивидуа у три димензије. 3-D систем се користи за реконструкцију 3-D слике. Име овог система је Light Coding.

Kinect сензор представља уређај правоугаоног облика повезаног са малим постољем и моторизованим деловима који омогућавају промену угла нагиба сензора. Дизајниран је да буде постављен испод или изнад дисплеја. Уређај се састоји од класичне РГБ камере, дубинског

сензора, низа микрофона и инфрацрвеног пројектора, који омогућавају комплетно тродимензионално праћење објекта, препознавање лица и препознавање гласовних команди. Уређај је у стању да прозна и издвоји глас из позадинске буке у кућним условима и тиме омогући да се видео позиви изводе без употребе слушалица или не сметано управља апликацијом. Тренутно подршка обухвата десетак светских језика.



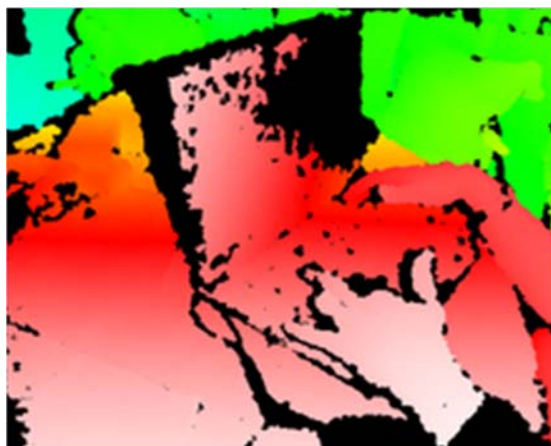
Слика 2-6 Структура сензора

Дубински сензор се састоји од инфрацрвеног ласерског пројектора комбинованог са монохроматским CMOS сензором који снима видео материјал у 3-D у затвореним просторијама. Опсег препознавања је променљив и софтвер је у стању да аутоматски калибрише сензор на основу ситуације и физичког окружења, што је изузетно корисно за ситуације када је присутан намештај или друге препреке.

Софтверска технологија је омогућила поправку начина праћена напредних гестова, препознавања лица или гласа. Kinect је у могућности да прати до шест корисника истовремено од тога два корисника могу активно да учествују у коришћењу и буду праћена анализом покрета која прати двадесет тачака па особи. Ово ограничење је заправо софтверско јер уређај је у стању да прати онолико особа колико је у његовом видном пољу.



Слика 2-7 Инфрацрвена слика показује ласерску мрежу коју уређај користи да израчуна дубину



Слика 2-8 Дубинска мапа је искоришћена за преливање боје од црвене (близу) до зелене (далеко)

Инверзним инжењерингом је утврђено да Kinect има различите видео излазе којима фреквенција иде у распону од ~ 9 Hz до 30 Hz у зависности од резолуције. Стандардни стрим користи 8 битну VGA резолуцију од 640x480 пиксела са Bayer филтером боја али хардвер је способан да снима до 1280x1024 на нижим фреквенцијама. Монохроматски дубински видео је резолуције 640x480 са 11 бити дубине, видео прави 2048 нивоа осетљивости. Kinect је у стању да прикаже и директан приказ онога што инфрацрвена камера види са ове две претходно поменуте резолуције.

Сензор има практичну дистанцу са које може да прати покрете у распону од 1,2-3,5 метара за уређај који се користи за Xbox 360. Област која је препоручена за коришћење сензора је негде око 6 метара квадратних. Сензор може да прати са продуженим режимом на даљини од 0,7-6 метара.

Kinect има угао видљивости од 57° хоризонтално и 43° вертикално. Мотор у постољу омогућава вертикалну промену угла до 27° у оба правца. Због угла гледања и минималне удаљености од сензора може се прерачунати да је 1,3 mm еквивалент једном пикселу.

Низ микрофона се састоји од четири микрофонске капсуле и управља са сваким каналом са 16 битним звуком са семплинг фреквенцијом од 16 kHz.

Зато што Kinect моторизовани механизам за промену угла захтева више енергије него што класичан порт на Xbox 360 може да обезбеди, уређај користи додатни конектор који комбинује енергију коју добија са Xbox 360 и из посебног напајања. Нови Xbox 360 уређаји имају посебан порт који служи за ову сврху док старије верзије захтевају још један прикључак за струју.

2.2.1.2 Kinect за Windows

Microsoft је фебруара 2011 године објавио да ће издати не комерцијалу верзију Kinect SDK за оперативне системе. Ова верзија се појавила јуна 2011 године за Windows 7. SDK је садржао драјвере компатибилности за Kinect уређаје и могућност програмирања у већ споменутим технологијама. Поред тога садржао је следеће могућности:

- Сирове информације са сензора, укључујући све сензоре.
- Праћење скелета. Праћење скелета две особе које се налазе у видном пољу са по 20 тачака.

- Напредне аудио могућности. Утврђивања који скелет говори тренутно и све већ споменуте могућности уграђене у Windows speech recognition API.
- Комплетну документацију и примере кода употребе.

2.2.1.2.1 Kinect за Windows 1.5

Kinect за Windows 1.5 SDK је садржавао и 'Kinect Studio нову апликацију која је дозвољавала програмерима да снима, приказу и дебагирају клипове корисника који врше интеракцију са апликацијом.

Такође уведен је нови "седећи" мод који је у стању да са веома мале удаљености прати главу, руке и врат корисника у 10 тачака. Наравно овај мод ради и у ситуацији када корисник седи или стоји близу.

Уз знатно ширу подршку за језике, ова верзија SDK је учинила Microsoft Kinect да се одвоји од конкурентских производа и наметне себе као технологију коју смо одабрали за наш практичан пример.

2.2.2 Softkinect

Softkinect представља конкурентски производ Microsoft Kinect-у истоимене белгијске компаније. Овај сензор спада у категорију time-of-flight (TOF) камера. Које су прикладне за препознавање објеката у два мода, ближи (15cm до 100cm) и даљи (150cm до 400cm). Намењен је употреби у затвореном простору. Камера долази са DepthSense SDK и API-јем који дају увид у дубинске и RGB податке. Доступан је и за Windows и за Linux.[6]



Слика 2-9 Softkinect сензор

Основна конфигурација сензора ове врсте је за све производе иста, основна разлика је у перформансама и софтверу који их прати.

Сензор је састављен од следећих компоненти:

- Резолуција дубинског сензора: QQVGA (160x120)

- Резолуција сензора у боји: VGA (640x480)
- Дубинско видно поље: 57.3° x 42.0° x 73.8°
- Извор дубинског осветљења: LED
- Уграђени микрофони: Два
- Струја и повезивање са рачунаром: USB 2.0 и спољно напајање
- Димензије: 24cm(ширина) 4cm(висина) 5cm(дебљина)

Уз сензор долази изузетно квалитетан софтвер који у потпуности прати Microsoft Kinect SDK.

SDK карактеришу следеће особине:

- Подржава велики број камера
- Могућност праћења четири корисника
- Библиотека гестова
- Ресурси и примери кода
- Праћење целог тела
- Две руке и прсти праћени у блиском моду
- Праћење гласа
- C++ API, C# wrapper и поједностављен API
- Додатак за Flash

Ова камера има својих предности, док Microsoft Kinect има својих, одабир сензора за сад је условљен могућношћу куповине Kinect сензора који је много доступнији. Крајњи избор зависи од захтева апликације и пословних одлука аутора будуће апликације.

2.3 Кориснички интерфејс орјентисан ка NUI

У поглављу 2.1 смо видели предности NUI-а а сада ћемо објаснити шта заправо значи и како се креира природан интерфејс.

Важно је основне концепте природних интерфејса укључити тако у процес размишљања да постану интегрисани део. Јер када се прича о некој специфичној технологији, као што је multi-touch или препознавање покрета, потребно је применити концепт природног људског понашања на њу инстинктивно.

Као што је поменуто основни концепт је коришћење основних способности и вештина, и у наставку ће бити показано пар примера како људске способности можемо применити на кориснички интерфејс.

Урођене способности се развијају одрастањем и у великој мери их користимо чак и да нисмо свесни тога. Разматраћемо како људско поимање трајности објекта можемо да искористимо у NUI-у. Размотрићемо како трајност објекта можемо да применимо у систему управљања фајловима. Након тога ћемо погледати шта је другачије у односу на то како GUI отвара и контролише фајлове и како ово може створити проблеме приликом фокусирања на објекат или креирање течне трансакције.

2.3.1 Постојаност објекта

Трајност објекта је когнитивни развојни основни корак где дете разумева да објекат и даље постоји и ако га није могуће даље видети, чути или додирнути. Ово је основни концепт који се

развија у прве две године живота и кључан је за разумевање да објекти су одвојени од њега и да су трајни. Деца која ово разумеју ће тражити објекат тамо где су га задњи пут видела.

Ово разумевање је у потпуности аутоматско једном када "сазри". Одрасли немају никакав когнитиван замор питајући се да ли ће њихова кола престати да постоје уколико их паркирате и уђете у зграду где их не можете видети. Једноставно разумете да можете наћи објекат тамо где сте га задњи пут оставили. У ситуацијама када нешто није тамо где сте га оставили изазива огромну дозу стреса. У великом броју случајева ваша меморија је крива и заборавили сте где сте нешто оставили. Када нешто није тамо где смо мислили, ремети се наше поимање постојаности објекта и ми морамо потрошити време и напор да решимо конфликт између очекивања и стварности.[3]

Можемо применити постојаност објекта на развој интерфејса тако што ћемо се осигурати да не прекршимо корисничко природно очекивање да је садржај трајан и да остаје у условима у којима је остављен.

Са NUI мењамо основне принципе како вршимо интеракцију са рачунарима. Али иза графике и даље морамо да се позабавимо истом архитектуром оперативног система и компјутерских апстракција. Концепти као што су драјвери, процеси и трендови, фајлови и систем фајлова ће бити ту још дуго.

Потребно је да размотрити неке од изазова који постоје са системом менаџмента фајловима и погледати како се може применити трајност објекта на овај проблем и како је могуће сакрити комплексност од корисника и да учинити њихово искуство природнијим.

2.3.2 Промене у систему управљања фајловима

У регуларном GUI-ју када креирамо објекат као што је Word документ, ми морамо да бринемо о снимању документа као фајла, како да назовемо документ и у који директоријум ћемо да снимимо фајл. Имплементација оваквог фајл менаџмента има неколико проблема уколико хоћемо да га применимо на NUI.

Управљање фајловима у GUI-ју је композитна вештина и потребно је одређено време да би се савладала. И ако фајл и директоријум имају корене у стварном животу, постоје много начина управљања фајловима која нису природна. Као прво, уколико заборавимо да снимимо наша документа они могу да буду изгубљени трајно. Ризик заборављања је редукован кроз дијалог са упозорењем, или тако што неке апликације имају опцију аутоматског снимања, ова опција је конфигурисана да ради најчешће на пет минута.

Почетници у раду на компјутерима не разумеју начин функционисања фајла и фајла система и они најчешће уче једино кроз фрустрацију губљења документа. Кроз NUI имамо могућност да учинимо корисничко искуство бољим и за почетнике и за напредне кориснике. Размотрићемо имагинарну апликацију како би илустровали како се користи трајност објекта да учимо систем управљања фајловима природнијим. Овај пример назовимо "Природни-документ".

Апликација је креирана за употребу на multi-touch таблет уређају и дозвољава вам да креирате документ неког типа и решава цео проблем управљања фајлом уместо вас. Аутоматски се снима сваки пут када нешто промените на фајлу и сама одређује како да назове фајл и где да га складиште. Све о чему корисник мора да брине је да креира документ и да га користи. Овај начин понашања апликације изузима све оне непотребне ствари које корисник не мора да зна.

Апликације типа као што су Microsoft OneNote прате овај шаблон рада. У њима корисник хвата белешке и у било ком тренутку може да затвори апликацију. Касније када отворите апликацију ваше белешке су тамо тачно где сте их и оставили. Ово је прави пример постојаности објекта јер да би објекат заиста био ту и био постојан о не сме нестати након затварања апликације.

2.3.2.1 Отварање и затварање фајлова

У стандардној GUI апликацији, када затворите апликацију садржај нестаје заједно са прозором апликације. Уколико хоћете да му приступите поново морате да га пронађете и да двапут кликнете на одређену икону, правилно позиционирану у хијерархији фолдера која показује на место где је апликација смештена на диску.

Овај стил отварања докумената је веома ометајући појму постојаности објекта. У једном моменту садржај је приказан у пуној величини а касније је приказан у виду иконице на тотално другом месту са другачијим начином изгледа. Постоје само логичка повезаност а не визуелна између ових објеката. То је оно што нови корисници имају проблема да науче.

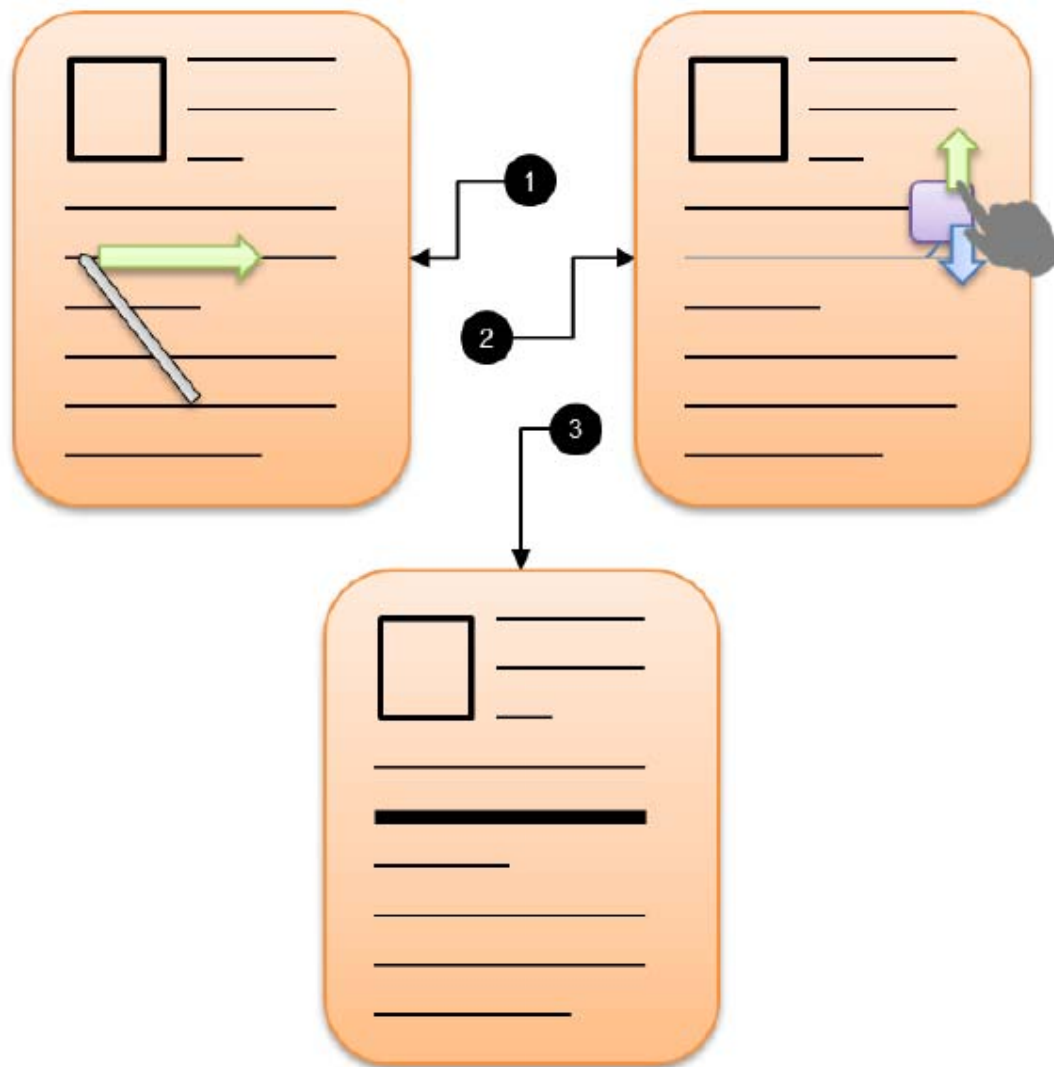
Други начин учествовао у GUI је да се отвори апликација а онда да се садржају приступи отварању фајла из апликације. Или преко навигације кроз хијерархију докумената или преко скорије отварањих фајлова. Када покрећемо апликацију ми заправо кажемо рачунару који део кода је потребно да изврши да би нам нешто приказао, исто се дешава и са фајл системом. Ово представља начин како раде рачунари али заправо ово нису информације које корисник мора да зна.

Код графичког интерфејса учестало је да корисник мора да брине о томе да ли апликација ради. Све што корисник би заправо требало да ради је да се посвети садржају који је желео у том моменту. Ако би "Природни-документ" кренуо корак даље и покушао да направи корак даље ка решавању ових брига. Прво ћемо имплементирати директну интеракцију а онда постојаност објекта.

2.3.3 Интерфејс оријентисан ка садржају (Content-Centric UI)

Једно од основних начела природних корисничких интерфејса је "директна интеракција са садржајем", што би буквално значило да у неком моменту је потребно "заборавити на апликацију" колико год то чудно може деловати. NUI није оријентисан ка интеракцији са апликацијом, менијем или неким другим делом интерфејса осим садржајем. Применићемо смернице директног управљања да "Природни-документ" постане апликација оријентисана ка садржају. Оријентисаност ка садржају заправо значи да нема апликације тј. интерфејса апликације, наравно у позадини биће доста процеса и тредова али о томе корисник не брине.

За "Природни-документ", сваки документ који корисник креира или погледа биће одвојени објекат. Свака ставка садржаја је један објекат. Ови објекти садрже визуелни приказ садржаја као и неопходне елементе корисничког интерфејса за интеракцију са садржајем. Елементи интерфејса ће бити контекстни што је више могуће. Могуће је минимизирати број елемената у интерфејсу тако што ће се они појављивати само када је одбран садржај одређеног контекста. Учестале интеракције дозвољавају кориснику да се брже помера са једног задатка на други навигацијом кроз контекстни интерфејс и изводећи акције. Следећа слике приказује један шаблон интеракције за промену величине фонта текста и ово је добар пример директне интеракције.



Слика 2-10 Пример промене величине слова директном интеракцијом

Један начин промене величине слова је тај што контекста интеракција почиње превлачењем прста преко жељеног текста. Ово ће изазвати да се текст обележи и да се појави мени прикладан за манипулацију прстом. Потезом прстима који симулира увећавање фонт ће се повећати прикладно гесту. Селекције се истовремено може искористи и за друге промене као што су премештање текста или промена стила. Дакле једна селекција није везана за једну операцију.

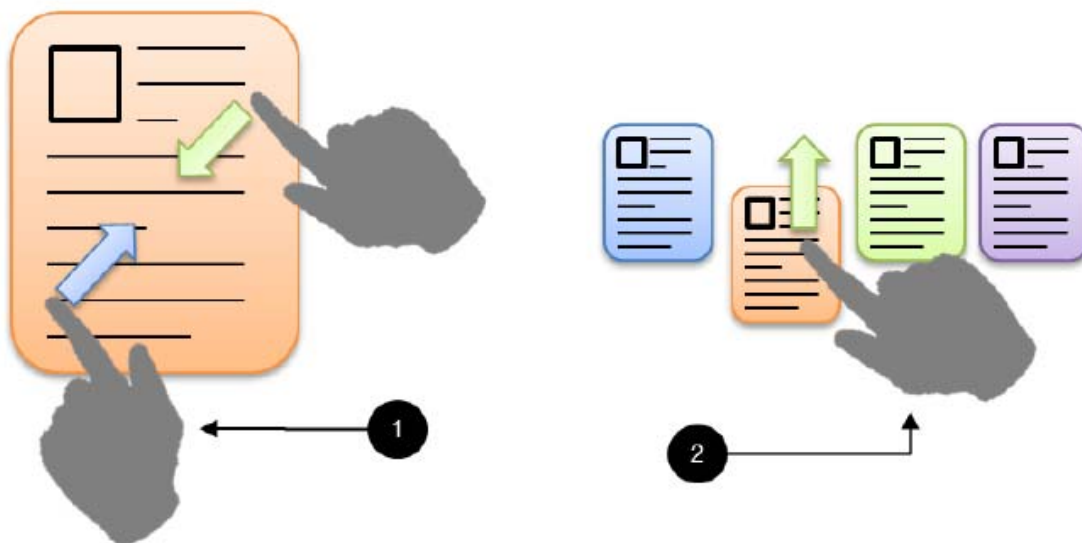
Напомена:

Иако корисник спроводи гест на самом тексту, користићемо помагање приликом баратања да кориснику омогућимо да избегне предвиђање трансформације геста у манипулацију текстом. Постоји много других решења и ово је само пример где корисничким тестирањем треба изабрати најсуптилнији начин интеракције.

Чим се ова промена деси, апликација аутоматски снима фајл и позадини. И може се наставити са радом нормално а контекстни интерфејс нестаје чим се контекст промени.

2.3.3.1 Перманентни објекти захтевају течне трансакције

Ако попричамо о задњој интеракцији можемо дизајнирати и комбиновати оријентисаност ка садржају са трајношћу објекта. Када завршите са интеракцијом и документом уместо да затворите апликацију може се гестом прстима за скупљање и смањити цео документ, где ће приказ одвести на неку другу локацију приказану на следећој слици.



Слика 2-11 Завршетак рада са документом

Касније се корисник може врати на исто место и наставити рад на документу. Ова интеракција између активног на неактиван и поново на активан објекат може бити аутоматизована и активирана са прикладним гестом или елементима интерфејса за које је битно да се увек уместо њих може једноставна гест манипулација.

Скупљање и померање документа имплицира специјални начин да се складиште, организује и проналазе садржаји. Долазећи из GUI света, корисник је приморан да мисли о овоме као о замени за Desktop, али реално не постоје разлози зашто би ово било тако. Ова документа би могла бити садржана у оквиру области зумирања (Prezi окружење [7]), временске линије или било којег другог концепта интерфејса који има користи за овај садржај.

Овај специфичан опис за "Природни-документ" може или не мора бити најбољи приступ за праву апликацију, али најбитнији део је како смо узели природну способност коју сваки корисник има и искористили је за пример код неколико различитих примера креирања интерфејса и како су уклоњени непотребни детаљи који захтевају познавање како рачунар ради.

За "Природни-документ", садржај је трајан објекат и постоји течна транзиција између различитих стања објеката. Задржавањем тачности трансакција држимо корисника концентрисаним и смањујемо несигурност у раду. Уколико транзиција између стања није течна, корисник није у стању да интуитивно разуме како се може врати на претходно стање.

Како је трајни објекат у "Природном-документу" аутоматски снимљен и складиштен иза сцене. Ово минимизује број или комплексност вештина које су потребне новом кориснику да почне да користи апликацију.

Најбољи део код коришћења урођених способности како што је трајност објекта је што корисник не мора да научи ништа да би ово разумео. Објекат се понаша као што би се понашао и стварни објекат. Наравно не може се све у интерфејсима урадити на овај начин и ипак је потребно научити нешто од нових вештина. У наставку ћемо описати како се користе основне вештине у креирању интерфејса.

2.3.4 Коришћење основних вештина

Основне вештине је веома лако поново искористити у новој ситуацији. Што их чини савршеним за NUI . На пример, ако користимо вештину која се надовезује на трајност објекта. Једном када деца схвате да објекат не нестаје чак ни када га није могуће видети они су у стању да почну да комбинују ово са повезивањем односа и својом способности да ухвати и помере објекат. На тај начин настаје смештање објекта у објекат.

Повезивање односа је једна од првих ствари које разумемо као новорођенчад. Чим смо способни да ухватимо и померимо објекат, почињемо игру са повезивањем односа између објеката. Ово заправо преставља једну од главних занимација-игрица које дете само иницира. Деца обожавају да ставе играчку у корпу или кутији и онда пренесу целу кутију и изваде лопту. Учење деце да померају ствари где им је место је заправо одлагање објекта у одговарајући контејнер.

Ово вештине се могу применити на NUI у многим ситуацијама. Сада ћемо погледати како је ово могуће искористити за категоризацију објеката и поређење како се ово обаља у GUI класичном окружењу.

Замислимо активност у NUI апликацији где се врши разврставање објеката смештањем у њихов контејнер. Ово је веома једноставно и природно, зато што инстинктивно препознајемо повезаност објеката различитог типа и контејнера. Једном када схватимо да можемо да управљамо објектима померањем можемо лако препакovati објекте. На екранима осетљивим на додир ово може бити drag-and-drop операција. Са Kinect-ом интеракција ће бити тако што ћемо користити гест за хватање уместо додиривања тако да се може називати grab-and-drop.

Постоји неколико опција са GUI апликацијама за категоризацију објеката. Овде је наведено неко интерфејс патерна, од оних са већим когнитивним замором до оних са мало замора:

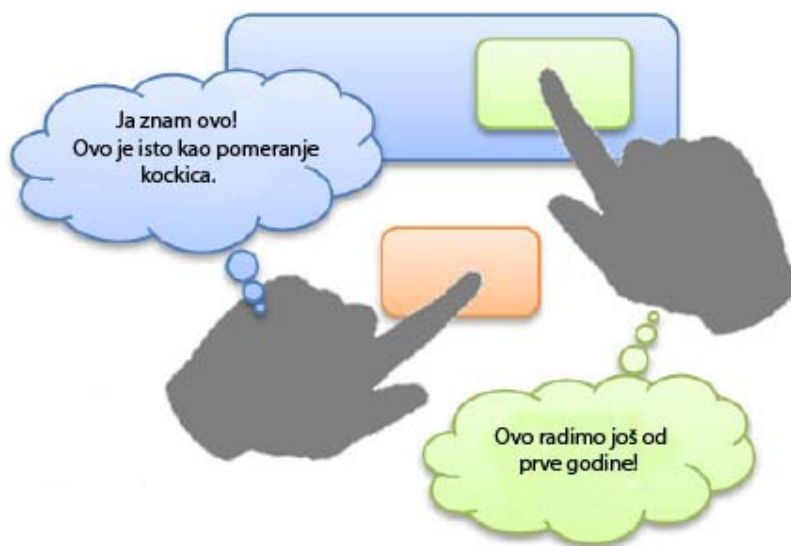
1. Листа објеката са пољима за чекирање, једно категоријска листа и дугме за примени.
2. Листа објеката са падајућом листом категорија по објекту.
3. Две листе, једна поред друге, са стрелицама на лево или десно које омогућавају промену релација.
4. Превлачење иконица са једне области и пуштање на другу област.

Прве три опције захтевају доста учења да би се контроле научиле и користиле као и сама интеракција са контролама. Последњи начин је скоро исти као drag-and-drop у NUI, нажалост веома мало је коришћен осим приликом премештања фајлова из једног директоријума у други.

Играње са контејнерима је нешто што свако од нас зна, али постоје компликованије компјутерске активности. Понекад је потребно научити вештину која је нова за коју не постоји тренутно научена као замена.

2.3.5 Учење нових вештина

Рецимо да смо препознали одређени гест или задатак који корисник треба да обави на интерфејсу а да није свакодневни људски гест. Како се додаје све више напредних могућности у апликацију ово ће се кад тад десити. Погледаћемо како да научимо једноставан гест који није тако природан за велики број корисника и не познају га од раније: Манипулација објекта (pinch) гестом са две руке.



Слика 2-12 Гест са две руке

Гест са две руке могуће је користи за манипулацију различитим објектима и начинима. У NUI он представља усвојени метод за промену величине објекта. Клише је промена величине слике. Такође ове гестове је могуће користи за промену бројне вредности. Раније је објашњени како апликација "Природни-документ" једноруком "pinch" гестом мења величину слова. Што заправо представља само промену броја.

Следећа табела показује шта је то што корисник може научити из ове активности.

Научена ставка	Опис
Гест увећања и умањења	Ово је заправо вештина додиривања екрана са два прста и померања објекта у односу на то. Ово је вештина која се лако учи и може се користи у различитим ситуацијама
Понашање интерфејса приликом промене величине слова	Разумевање понашања интерфејса у овим ситуацијама и препознавање реакције у сличним ситуацијама.

Табела 1 - Ствари које је корисник научио гестом

У овом случају овакви гестови су нешто што се може користити у многим другим ситуацијама. У апликацији "Природни-документ" искористили смо овај гест да би смо смањили документ и склонили га, као и за повећање неактивног документа и почетак рада на њему.

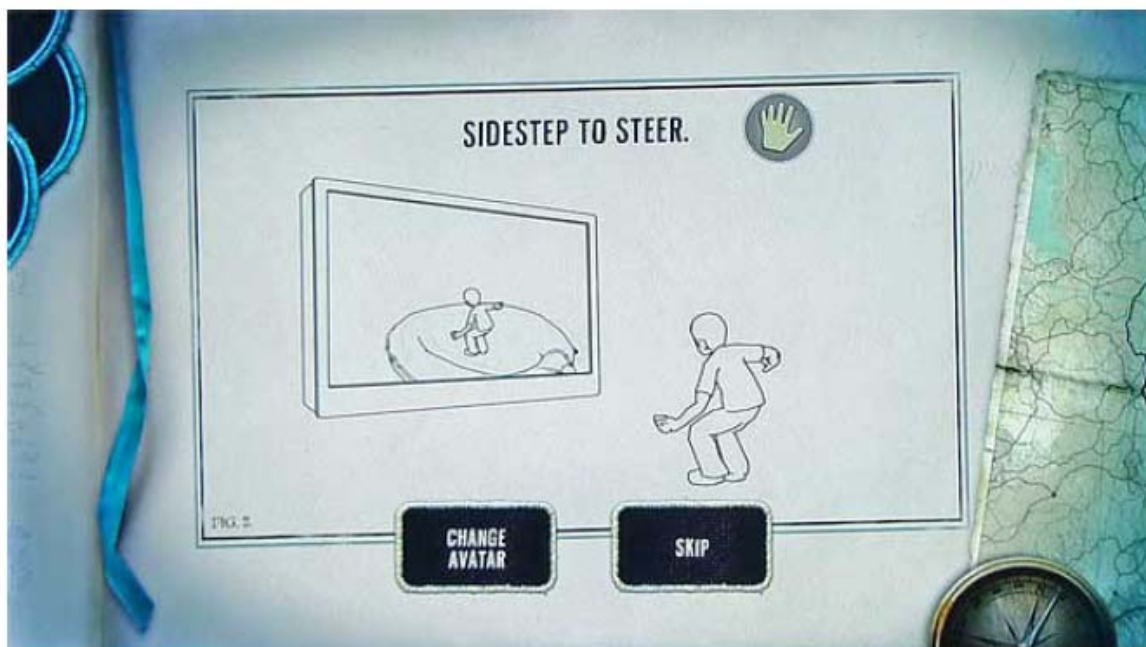
2.3.5.1 Учење вештине посматрањем

Постоји неколико различитих начина да се научи вештина интеракције. Уколико је једноставна вештина, најједноставнији начин је да се посматра неко ко користи исту вештину. Извођена су тестирања над децом узраста од три године која су један минут посматрала неког одраслог како користи поменуте гестове за манипулацију мапама на табелу и затим се су деца пуштена да покушају да реплицирају понашање. У велико број случајева деца су успела да понове операцију што је било веома импресивно јер се њихова координација између руку и ока још увек развија.

За апликације које се користи у социјалном окружењу ово је добар начин да се научи како се користити интерфејсом, али ово не може бити једини начин учења. Понекад је нико није поред корисника да му покаже како се интерфејс користи.

2.3.5.2 Директна демонстрације вештине

Други начин учења вештине када посматрање другог корисника није могуће је да апликације демонстрира акцију. У multi-touch или Kinect апликацијама, ово може бити показано како сенке руку управљају интерфејсом. Алтернативно могуће је показати анимацију или видео неког другог како манипулише апликацијом. Ово је приступ који користе игрице на Xbox 360 Kinect играчкој конзоли. У неким Kinect, игрицама играчу је презентована анимирана демонстрација како се игрица игра док се игрица учитава.



Слика 2-13 Пример приказивања анимације за демонстрацију употребе игрице. Пример је узет из игрице *Kinect Adventure* за Xbox 360

Још један од учесталих начина рада Kinect игрица је што уколико корисник не игра правилно, игрица му приказује картунизовани дијаграм у горњем углу који показује акцију коју треба извести на правиан начин.

За разлику од игрица где овај вид демонстрације је веома добар, за апликације које се користе NUI у друге сврхе ово можда није најбољи приступ.

Један од проблема овакве демонстрације је одредити када је потребно пустити анимацију или видео. Видео игрице могу генерално да измере напредак корисника на основу неких меродавних параметара, али ово је веома тешко а у неким ситуацијама скоро и немогуће урадити у осталим апликацијама.

Неке апликације ће покренути демонстрацију уколико корисник није додирнуо апликацију за одређени временски период. Следећа слика показује Microsoft Surface финансијску апликацију са примером демонстрације на овај начин. Проблем са овим видом комуникације са корисником можда нема никакве везе са тим да је корисник збуњен и да не зна да настави даљи рад. Корисник можда једноставно ради нешто друго у овом моменту у стварном животу или са неким другим дискутује о томе што је приказано на екрану.



Слика 2-14 Демонстрација коришћења на Microsoft Surface финансијској апликацију

Још један проблем је демонстрације видеа или анимације која могу бити увредљиве кориснику зато што апликација мисли да корисник не зна шта треба да ради. Или корисник може помислити да апликација преузима контролу или да прекида нешто што је он већ радио. Овде је контекст знатно другачији од концепта у игрицама када корисник а и апликација јасно знају дали корисник добро поступа или не.

Скоро у сваком случају, постоје случајеви да се интелигентно дизајнира апликација да пригодно одговори на задатке учења корисника у овим ситуацијама. Уколико корисник не може да схвати како да изведе задатак играјући се са интерфејсом и морате да прибегнете приказивању демонстрације, велика је могућност да можете ако размислите да поправите интеракцију и интуитивност коришћења апликације. Најбољи приступ је примена прогресивног учења и креирања задатака који помажу учење.

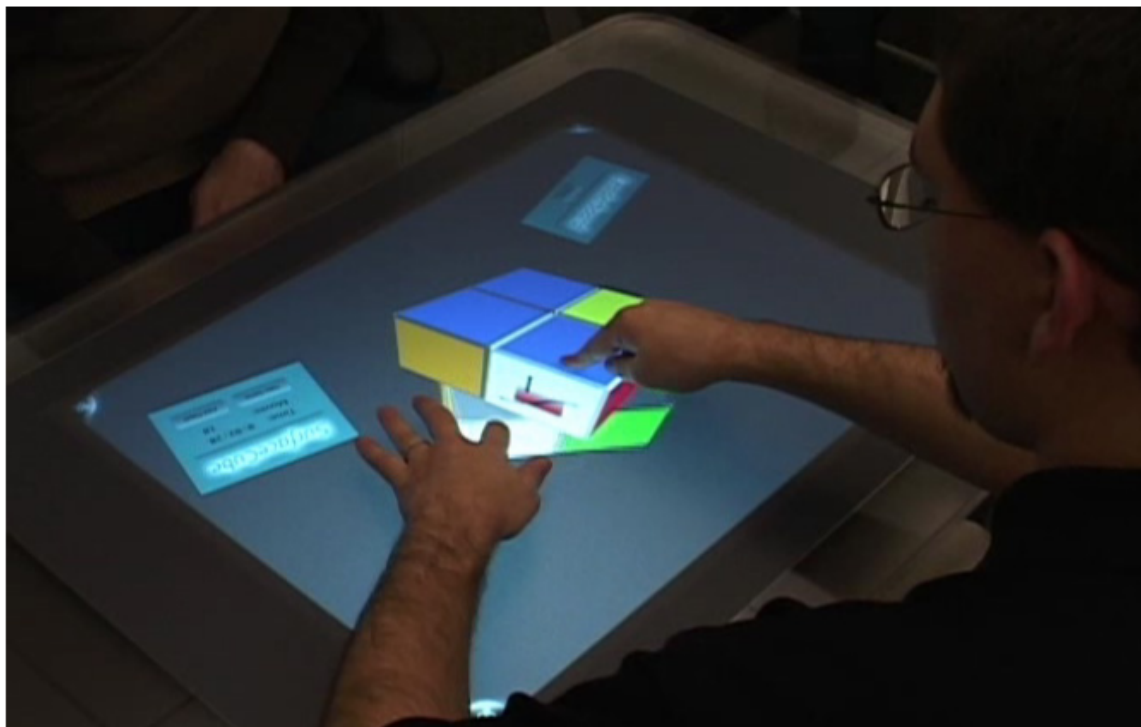
2.3.6 Задаци за учење употребе интерфејса

Задаци за учење су редовни задаци који имају за циљ да науче корисника специфично захтевану вештину или итерацијски шаблон. Добри кандидати за задатке за учење су они задаци тј. активности које нови корисници обављају у раној фази интеракције. Ово задаци не морају искључиво бити оријентисани ка учењу него и сналажењу.

Ови задаци могу да буду укључени у радни ток апликације да се би се постарали да корисник их пређе и научи. Ово у најчешће најбитније активности које корисник може радити у апликацији и потребно их је научити пре неког напреднијег рада.

2.3.6.1 SURFACECUBE

Као помоћ за боље разумевање задатак за учење, размотрићемо апликацију која је креирана баш у ове сврхе. Апликација се зове SurfaceCube, креирана је за Microsoft Surface уређај прве генерације. И у суштини представља Рубикову димензија 2x2. Има 3-D коцку којом је могуће манипулисати, ротацијом или померањем. Гест који се за ово користи зове се hold-and-flick. Овај гест се базира на томе да је потребно ставити прст на један део које желимо да остане непокретан и померањем другог прста ка страни којој желимо да ротирамо.



Слика 2-15 SurfaceCube апликација

Један од изазова приликом дизајнирања апликације је како научити кориснике да управљају 3-D објектима. Да би се ово урадило промењен је начин на који се стартује игрица. Приликом стартовања, кориснику су приказане две контроле. Један контрола је листа са резултатима претходних корисника. А друга је главни контролни панел који служи да подешавање нивоа тежине игрице и покретање. Обе контроле су приказане као ScatterView објекти.

Напомена:

ScatterView је WPF контејнер у оквиру Microsoft Surface SDK-а који омогућава садржају да може бити померан, скалиран и ротиран слободно као одзив на манипулације додиром.

Уместо да постоји "Start" дугме, аутор се одлучио да покретање игрице претвори у задатак за учење. Постоје две битне ставке које уче корисника:

- Могуће је окретати целу коцку класичним гестом.
- И могуће је окретати само један део Рубикове коцке користећи hold-and-flick гест.

Одлучено је да друга ставка постане начин покретања игрице и до доласка до главне коцке. Да би ово био задатак за учење избегнута је комплетна комплексност главног проблема и коцка нема све степене слободе и коришћена је мања коцка која има само две коцкице.



Слика 2-16 Начин покретања игрице кроз учење

Мини коцка може да се окреће само по једној оси за разлику од главне која може да се помера по обе осе. Поред тога ова коцка има само четири могуће конфигурације. Такође уводни алгоритам је написан тако да је покретање игрице тј. исправно решен проблем удаљен само једним гестом од решења.

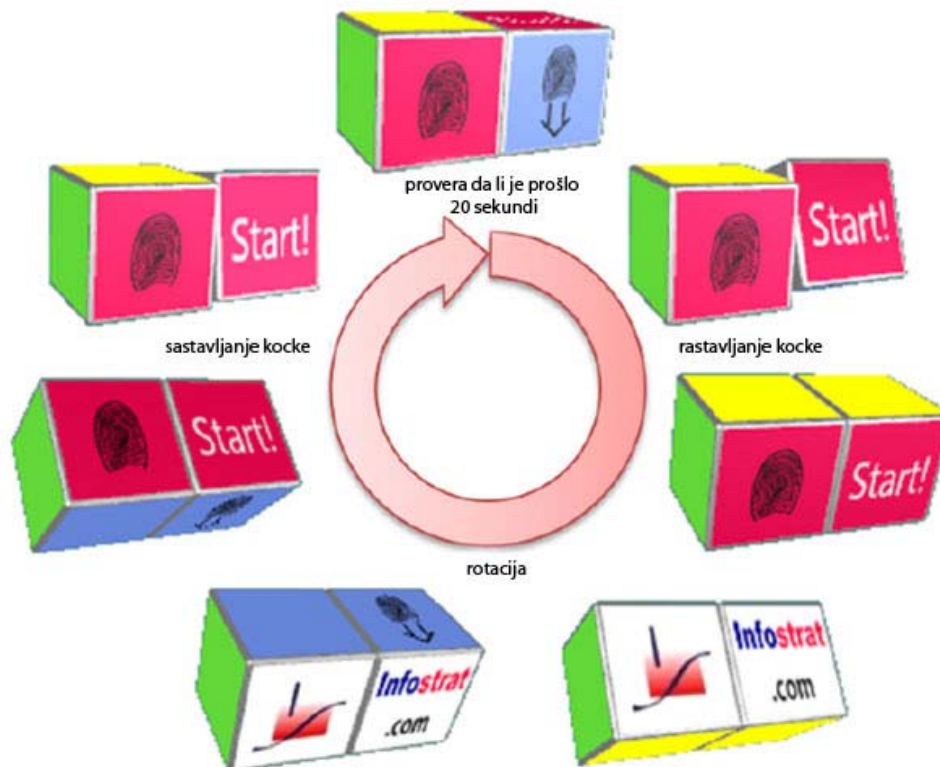
Следећа табела показује пет различитих техника коришћених за учење корисника шта коцка све може да ради и како је то могуће извести.

Техника	Имплементација	Опис
Једноставан задатак	Две коцкице са једном осом ротације	За разлику од главне коце која има осам коцкица и две осе ротације, мања коцка је једноставнија. Ово смањује број могућности о којима корисник мора да размишља и повећава шансе за успех.
Показати шта је могуће	Временска анимација	Уколико мини коца није додирнута, сваких двадесет секунди приказаше анимацију у три корака која траје једну секунду. Анимација показује могуће манипулације са коцком.
Иконографија	Иконице са отисцима прста	Иницијалне стране које су окренуте ка кориснику имају иконице на њима. Црвена страна има иконицу са отиском а плава страна са отиском и стрелицом на доле. Ово је начин да се кориснику наговести где је потребно да додирне и да примени кретање.
Визуелна интезивност	"Старт" текст на раздвојеним црвеним наличјима коцке	Коцка има окренуту једну страну ка кориснику али је друга страна такође видљива и њој пише Старт и има црвене боје, што подсвесно подиже тензију и подстиче на решавање проблема. Такође јасно наговештава да ће ако се коцка решити доћи до покретања игрице.
Текстуална обавештења	Текстуалне инструкције	Такође као додатну инструкцију имамо текст који каже: "Поклопите боје за старт"

Табела 2- Приказ задатак за учење пре покретања игрице

Временска анимација приказана на следећој слици је другачија од анимације споменуте у причи о финансијској апликацији. Ова анимација не користи силуете руку за демонстрацију задатка. Уместо тога једноставно привлачи сама пажњу и показује које манипулације су могуће са мини коцком, анимација је суптилна тако да не одскаче од остатка интерфејса. (Слика 2-17)

Једном када корисник реши мини коцку, интерфејс се препакује сам да направи места за главну коцу. Главна коца тада се постепено појави у решеној конфигурацији а затим се сама "растури" на ниво тежине које је подешен пре почетка игрице. Показивање промене конфигурације служи да кориснику такође покаже начин понашања коцке.



Слика 2-17 Анимација

Састављање свих коцкица правилно, учи корисника све ствари које је потребно да зна за манипулацијом игрицом али не ствара сметњу напредним корисницима које овакви задаци могу из иритирати у неким другим апликацијама или ситуацијама. Захтев да корисник изведе гест осигурава да ће знати да се снађе када игрица почне и да познаје принципе.

Мини коцка је сјајан пример како имплементирати задатак учења и како прогресивна крива учења може бити примењена. Такође посматран је пример како искористити основне способности и како коришћење инстинктивних способности може водити ка решавању проблема приликом дизајнирања апликације. Ове смернице је могуће применити у свакој апликацији која користи природне корисничке интерфејсе.

Основна начела креирања интерфејса приказана у овом поглављу имала су за циљ да покажу норме и принципе које треба применити, јер разноврсност апликација које користе NUI је огромна и није могуће показати на класичан начин поступак дизајнирања интерфејса. Заправо за примену NUI је потребно променити комплетан начин размишљања приликом дизајнирања апликација.

Пошто смо размотрили основне концепте NUI време је да погледамо на једном примеру у пракси како је ово спроведено.

3 АПЛИКАЦИЈА: ИНТЕРАКТИВНИ ИЗЛОГ

3.1 Постојећи проблеми рекламирања у изложима и решење проблема

Сведоци смо да велики број туристичких агенција и агенција за продају некретнина користи своје излоге на крајње традиционалан начин, затрпавајући површину излога огромном количином папира са огласима и понудама. На овај начин стварајући велику дозу забуне код потенцијалних купаца. Купац да би се заиста одлучио да посматра ове огласе мора заиста да буде заинтересован и спреман да потроши непотребно доста енергије и времена у тражењу понуде. Поред тога купци који на овај начин приступе куповини су већ постојећа циљна група која је у доброј мери спремна на куповину и сама агенција не добија нове муштерије него се конкуренцијом бори око једне исте групе купаца.



Слика 3-1 Како изгледају данашњи излози

Поред претходно споменуте опције, постоји још једна могућност употребе излога за пласирање информација а то је излог који у себи има екран неке средње величине на којем се врати презентација из Microsoft PowerPoint-а или неке сличне апликације. Истраживања су показала да је ово још мање корисна опција иако је на први поглед савременија. Да садржај је лакше променити али шта будући купци имају од тога што је излог вама лакши за модификацију? Апсолутно ништа! Ова опција као што је већ речено се показала као штетнија јер кориснику се драстично смањује количина информација коју може да добије у моменту. Купац би морао да проведе много времена да сачека да апликација направи цео круг са слајдовима ако жели да види нешто поново или ако није успео да прочита до краја. Број људи који је спреман да овако потроше време је са маркетиншке стране занемарљиво мали.

Дакле са једне стране имамо велики број непрегледних информација приказаних на веома не занимљив начин али које кориснику омогућавају ипак да претражи и пронађе оно што га занима и да не губи време са непотребним подацима. Са друге стране имамо комплетно модерно окружење које је од користи само ако сте се у правом тренутку нашли испред слајдова или имате

довољно времена да сачекате цео круг да се окрене. Не смемо овде заборавити да је просечно време задржавања по излогу свега тридесет секунди. Уколико не успете да заинтересујете купца за тај период они ће сигурно отићи. А презентација може трајати и неколико десетина минута или на лош начин мењати слајдове (преспоро или пребрзо). Што се сигурно не одговара свачијој брзини читања.

Решење се само намеће. Потребно је направити презентацију која на брз и интуитиван начин обезбеђује кориснику податке, заузима мање места и приказује јасне и једино корисне информације. Спој претходних начина приказивања делује као добро решење. Проблем је како креирати ову презентацију и како би она изгледала, поред тога особље које је запошљено у агенцијама нема ни вештине ни ресурсе за овакав подухват.

3.1.1 Решење проблема

Уз помоћ најсавременијих информационих технологија креирати интерактиван излог који ће омогућавати пролазницима да уз помоћ интуитивних интерфејса управљају приказаним подацима и прегледају понуду брзо и ефикасно дођу до корисних информација.

Интерактиван излог би требало да има један екран велике дијагонале складно уметнут у остатак излога, на коме се врши презентација садржаја. Садржај је приказан у виду интерактивне апликације креиране преко природних корисничких интерфејса. Управљање апликацијом се врши праћењем покрета и гестова. Поред NUI коришћена је надograђена стварност која има за циљ побољшање ангажованости корисника, повећање заинтересованости и пружање једног новог угођаја које ће власнике овог излога издвајати од конкуренције.

За технологију праћења покрета одлучено је да се користи Microsoft Kinect а не површине осетљиве на додир. Из више разлога:

- Овим уређајима је могуће пратити већу површину истовремено.
- Уређаји осетљиви на додир великих површина су изузетно скупи.
- Могућност од физичког оштећења излога је сведена на минимум јер је сва техника смештена унутар радње.
- Могућност привлачења пролазника излогу чак и ако нису близу или га не посматрају тренутно.

Агенти продаје у компанији имају комплетне статистичке информације о претраженим локацијама и броју посетиоца. Могућност једноставног администрирања садржаја кроз клијентску апликацију.

Излог је замишљен као аутономни производ, потпуно не зависан, уклопљен у мали информациони систем који треба да обезбеди већи број корисних информација купцима са једне и продавцима са друге стране.

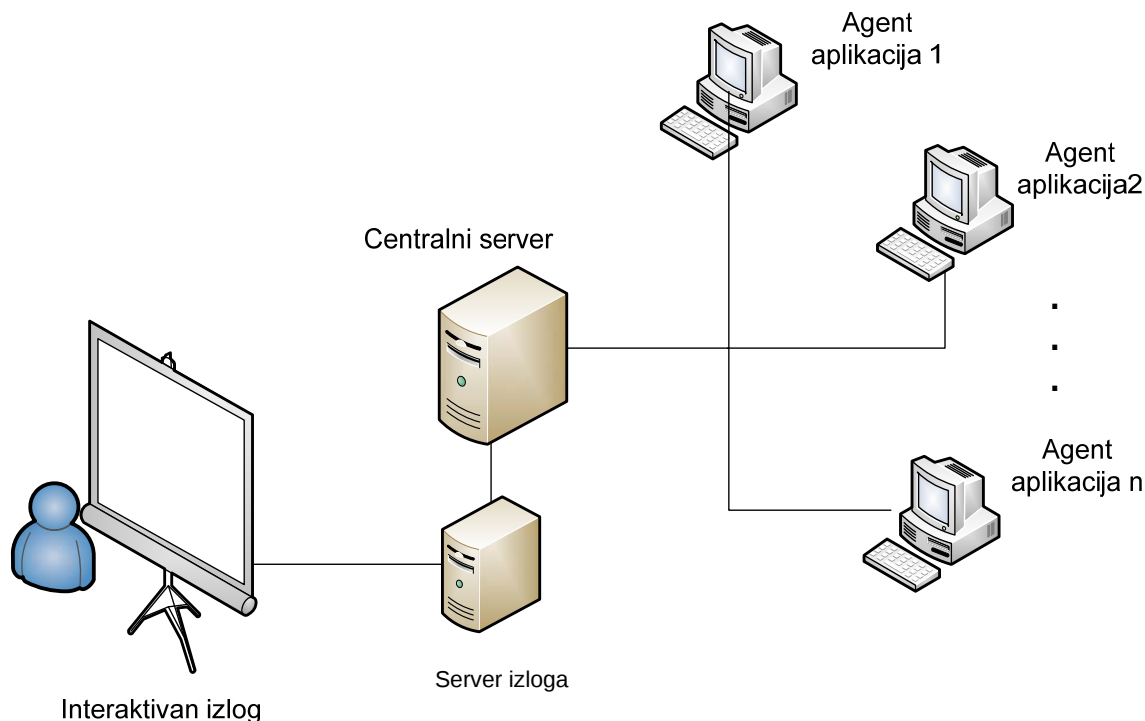


Слика 3-2 Интерактивне продавнице које се користе надграђеном стварношћу

3.2 Конфигурација ИС

Информациони систем је осмишљен да одговара конфигурацији приказаној на шеми 3.3. Састављен је од следећих компоненти:

- Централни сервер - Представља сервер базе податка који има задатак да обезбеди комуникацију излога са базом. Прикупи и анализира све податке потребне за статистику и комуницира са клијентским апликацијама.
- Сервер излога - Покреће апликацију интерактивног излога, комуницира са централним сервером и управља уређајима за приказивање и контролу покрета.
- Интерактиван излог - Састоји се од дисплеја који приказује интерактивну апликацију и уређаја за праћење покрета.
- Агент апликације - ове апликације за разлику од осталих компоненти нису хардверске, имају могућност администрације садржаја интерактивне апликације. Прегледања и извршавања статистичких анализа маркетиншких кампања спроведених на интерактивном излогу. Број ових апликација није ограничен структуром система већ искључиво постојећом инфраструктуром купца. Рачунаре на којима се ова апликација извршава обезбеђује купац.



Слика 3-3 Шема информационог система

Централни и сервер излога се морају налазити у локалној мрежи са интерактивним излогом без могућности удаљеног приступа осим у сврху администрације софтверских компоненти. Агент апликације могу и не морају бити у локалној мрежи јер њихова комуникација се обавља преко клијента који комуницира са Web сервисом преко локалне или спољашње IP адресе. У наставку ће бити детаљно објашњена свака појединачна компонента информационог система.

3.3 Агент апликација

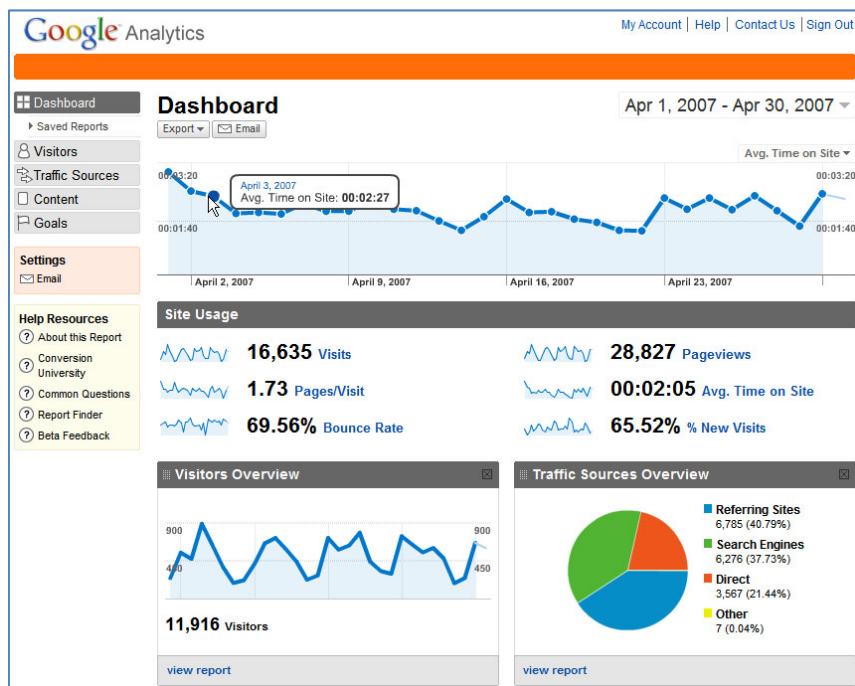
Проблем малих и средње великих туристичких агенција је то што нису у стању да се рекламирају на великим и скупим медијима и добијају клијенте на овај начин. Зато је потребно да се боре са локалном конкуренцијом за веома уску циљну групу. Апликације и комплетно решење интерактивног излога је право решење за њих јер одговара њиховом маркетиншком буџету.

Интерактиван излог представља нови маркетиншки канал који имај за циљ да повећа циљну групу привлачењем нових корисника тако што ће додатно заинтересовати неутралну корисничку групу, пролазнике. О начину рада излога и комуникацији детаљније ће бити говора у наредном поглављу.

Агенти који су задужени за продају и креирање маркетиншке стратегије могу одређивати шта ће се пласирати од понуда и информација у интерактивном излогу и одређивати шта је то што пролазници прво виде и шта желе да постигну. Најчешће је то продаја али може се говорити и о додатној промоцији неке активности или догађаја или о појачавању утиска јавности о агенцији као бренду.

Агент апликација представља администраторску апликацију нижег нивоа која дозвољава промене садржаја излога али са њом се не могу вршити било које озбиљне измене информационог система. Апликација може да креира/избрише било који део садржаја на крајње једноставан начин који је кориснику добро познат. Пошто администраторске апликације нису најпогоднија област за примену природних корисничких интерфејса овде ће више бити речи о начину рада и примени апликације него о самој апликацији.

Поред адаптације садржаја агенти могу користити напредне статистичке модуле, који у великој мери подсећају на анализу алата резултата посете интернет страница. На основу статистичких података могуће је поправити и боље одредити маркетиншку стратегију ради веће ефикасности.



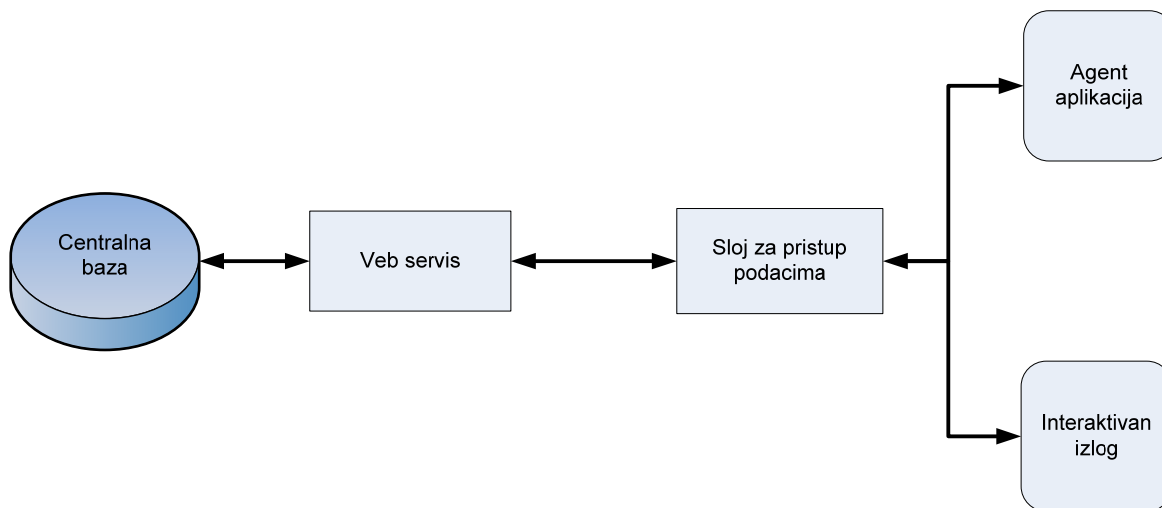
Слика 3-4 На овој слици приказан је пример алата за анализу посета одређеној интернет страници.

Наравно статистички модул наше апликације није толико свеобухватан као алат који је претходно споменут али је могуће прегледати велики број релевантних информација. Претходни пример је искоришћен као репер приликом креирања корисничког интерфејса јер као што смо раније већ говорили начин препознавања и сналажења у корисничком интерфејсу је у великој мери поприлично завистан од већ научених шаблона понашања. А пошто је ово веома специфична врста алата корисници који не припадају информатичкој струци најбоље ће се снаћи у окружењу сличном њима добро познатим алатима.

3.3.1 Слој за приступ подацима

Слој за приступ подацима је општа компонента информационог система користе га и излог и агент апликација. Пошто се прво упознајемо са агент апликацијом потребно је упознати се и са овим делом информационог система.

Слој за приступ подацима није приказан на физичкој шеми И.С међутим на следећој слици јасно се може видети његова позиција.



Слика 3-5 Ток података у информационом систему

Ток података је наравно двосмеран и креће се на релацији: Централна база података → Веб сервис → Слој за приступ подацима → Корисничке апликације.

Слој за приступ подацима представља такозване клијент апликације које имају за циљ да унифицирају и обједине начин баратања подацима. Основна намена је да свака корисничка апликација комуницира са клијентском апликацијом а она комуницира са веб сервисом. На овај начин поједностављује се развој апликација на дужи рок јер овај слој представља библиотеку коју само користе друге апликације. Зато није потребно поново развијати овај део, на тај начин се избегава редундантност кода, смањује обим тестирања апликација, штеди на броју инжењерских сати што додатно смањује цену развој тј. повећава профит компаније.

Веб сервис се користи из истих разлога као клијентска апликација али његова примена је додатно битна. У ситуацији када се апликација користи само из локалне мреже слој за приступ може бити довољан али се опет практикује израда сервиса. У било ком тренутку је могуће систем пребаци да ради са локалне на глобалну мрежу, директан приступ серверу базе података и самој бази на

мрежи би могао бити веома опасан и по безбедност информација и по "здравље" оперативног система. Зато је неопходно користити веб сервисе где се у само једној промени конфигурационог линка систем може пребацити из једне у другу мрежу.

3.3.2 Статистички модули и приказивања информација

Статистички модул представља "engine" задужен за вођење евиденције о догађајима који се дешавају у излогу. Свака интеракција корисника пропраћена је адекватном реакцијом система и бележењем у одређену статистичку категорију. Ове податке систем затим процесира и агенти добијају комплетну статистику њихове рекламне кампање.

Интеракције које систем може да прати су:

- Долазак новог корисника
- Отварање понуде
- Отварање категорије
- Време проведено на понуди
- Одлазак корисника.

Овај мали број догађаја може да обезбеди изузетно велику количину података које можемо приказати о интеракцијама корисника са излогом. Статистички подаци који су доступни су:

- Број људи који је посетио нашу кампању у задатом временском интервалу. Временска јединице могу да буду један дан, месец или година.
- Време задржавања корисника у апликацији.
- У које доба дана је апликација највише била посећивана.
- Статистика о понудама, најпосећеније понуде, не посећене, које врсте понуда су имале најбољи одзив.
- Колико је одиграно квизова - на основу овога агенти могу видети колико се корисника одлучило да након квиза уђе у агенцију.
- Однос између корисника који су користили апликацију и оних који су одиграли квиз.
- Колико понуда просечено се отвара по кориснику. Шта је то што гледају људи који се задржавају краће а шта гледају људи који дуже користе апликацију.

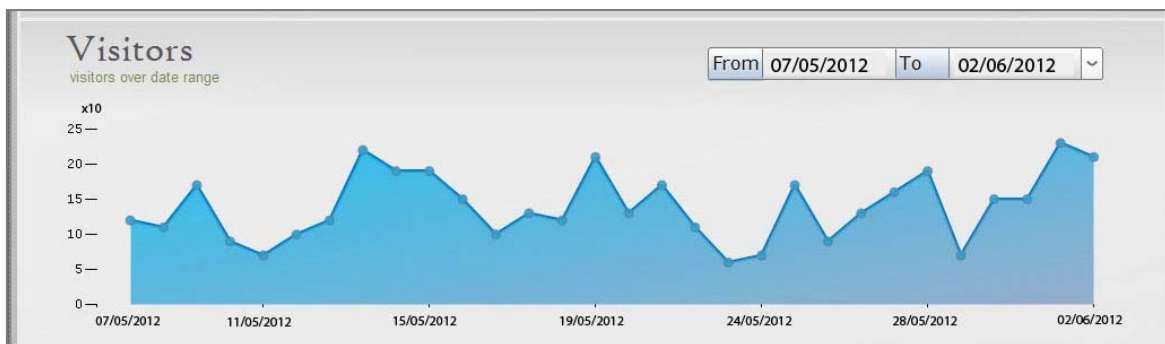
Апликација поред овога све статистичке податке може да изведе у Excel тако да практична примена информација зависи само од способности агента који њима барата.

Претходно наведени статистички подаци су подељени у следеће групе ради лакшег приступа:

- Понуде
- Корисници
- Времена
- Категорије
- Квизови
- Општа статистика - садржи најбитније податке из сваке категорије.

Сигурно једне од најбитнијих информација јесу оне о броју људи који су посетили апликацију. Прво велики број људи који користи апликацију није припадао групи људи који су били претходна циљна група, затим могуће је пратити "reach" кампање на укупну популацију која представља нову циљну групу. Подаци који приказују број посета за задати временски период се могу видети на следећем дијаграму. Такође доступне су информације ког дана је био максимум и минимум

посета и колико је износио. Може се пратити да ли се број људи повећава, опада или осцилује и како наставити у најповољнијем смеру.



Слика 3-6 Дијаграм броја корисника за задати временски период

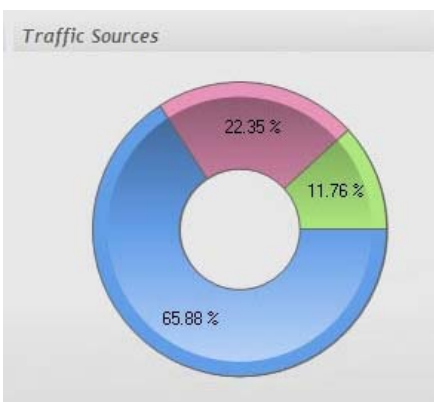
Време задржавања корисника показује да ли су особе пришле и користиле апликацију само зато што је нешто ново и што их је привукло или зато што су остали да сазнају неку корисну информацију, њима занимљиву.

Такође на основу доба дана када је апликација била најпосећенија можемо закључити који ти људи највише гледа наше апликације, поготово ако ово укрстимо са понудама које су прегледане у том временском интервалу. Такође на основу ових информација можемо прилагодити стратегију приласка и понуде корисницима.

Понуде нам могу показати које су наше најзанимљивије понуде, шта је то што људи желе да виде или где желе да путују итд. Такође постоји ранг листа базирана на процентуалном приказују прегледаних понуда која се односи на укупан удео. Такође могу се добити и информације колико корисник просечно отвори понуда, колики је минимум а колики максимум.

Информације о броју одиграних квизова показују колико су корисници заправо заинтересовани, колико је кампања била ефикасна и наравно колико је корака ка продаји нових аранжмана учињено. Ипак крајња поента апликације је повећање продаје.

На крају имамо информације о прегледаним категоријама типа: Летовања, Шпанија, Излети итд. Ове информације се приказују Pie графикомом и на основу њих могуће је утврдити шта треба променити у понуди и повећати продају или бар задржавање корисника у апликацији.



Слика 3-7 Број посећених категорија

3.4 Излог

Апликације излога је замишљена као: Интерактивно покретом контролисано решење излога које претвара не искоришћене стаклене површине у визуално примамљиве и увек доступне информативне штандове.

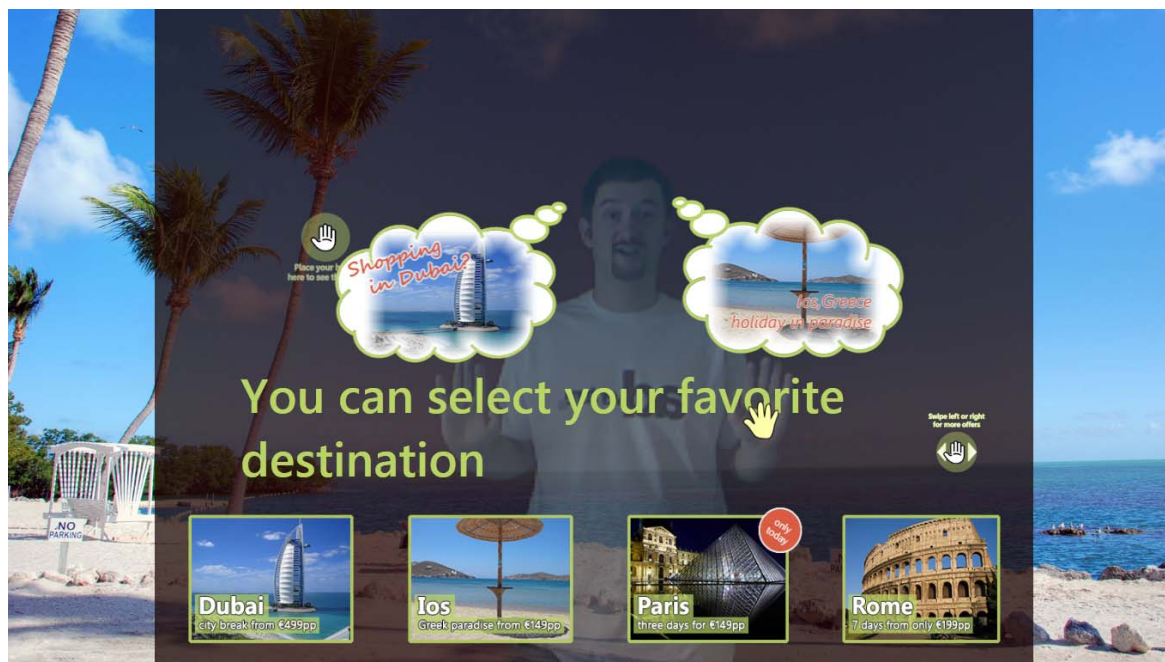
Излог чини монитор резолуције 50" или пројекција на велику површину путем пројектора који могу да емитују на веома кратим дистанцама. Величина платна онда је преко 70". Сервера за контролу излога и Microsoft Kinect уређај за праћење покрета.

Излог је замишљен да буде активан 99% времена. Што значи да апликацију није потребно гасити ради конфигурације, поновног подешавања, ре-сетовања или других разлога.

Сведоци смо да у свакодневном животу смо бомбардовани рекламама са свих страна: радио, телевизија, билборди, рекламни панои, мобилни телефони, флајери, презентери и многи други начини. Због тога смо еволуирали или боље речено научили да избегавамо и игноришемо ове рекламе. И то радимо тако добро да нисмо свесни да могу и информације које су нама корисне, није свако оглашавање непотребно, да нам промакнуту.

Интерактиван излог нуди нови вид оглашавања, реч је о новом маркетиншком каналу који пружа услугу оглашавања али корисник одлучује колико ће да види. Наравно зона деловања је много мања него телевизија или новине али је утицај вишеструко већи.

Ако нико од пролазника у датим тренуцима не ради у апликације значи нема активног корисника, апликација емитује анимације понуда које су најзанимљивије или се издвајају по неком другом критеријуму. Она је визуелно примамљива и лако скреће пажњу. У моменту када пролазник приђе излогу да види "шта се то дешава", излог реагује на корисника и окружење се мења. Сада пролазник постаје корисник!



Слика 3-8 Тренутак почетка интеракције са корисником

Привлачење пажње - Awareness

Корисник може себе видети у радној површини, на начин веома сличан огледалу само што површина реагује на његово померање. Прво што корисник примети је да се на позицији његове десне руке исцртава симбола шаке који прати његово померање. На овај начин се испуњава први задатак учења и упознавања са апликацијом. Затим, се видео снимак користи за надграђену стварност, препознаје се глава корисника и у виду облачића се приказују дестинације које воде на понуде. Ове дестинације се мењају у одређеном временском интервалу а контуре прате корисника.

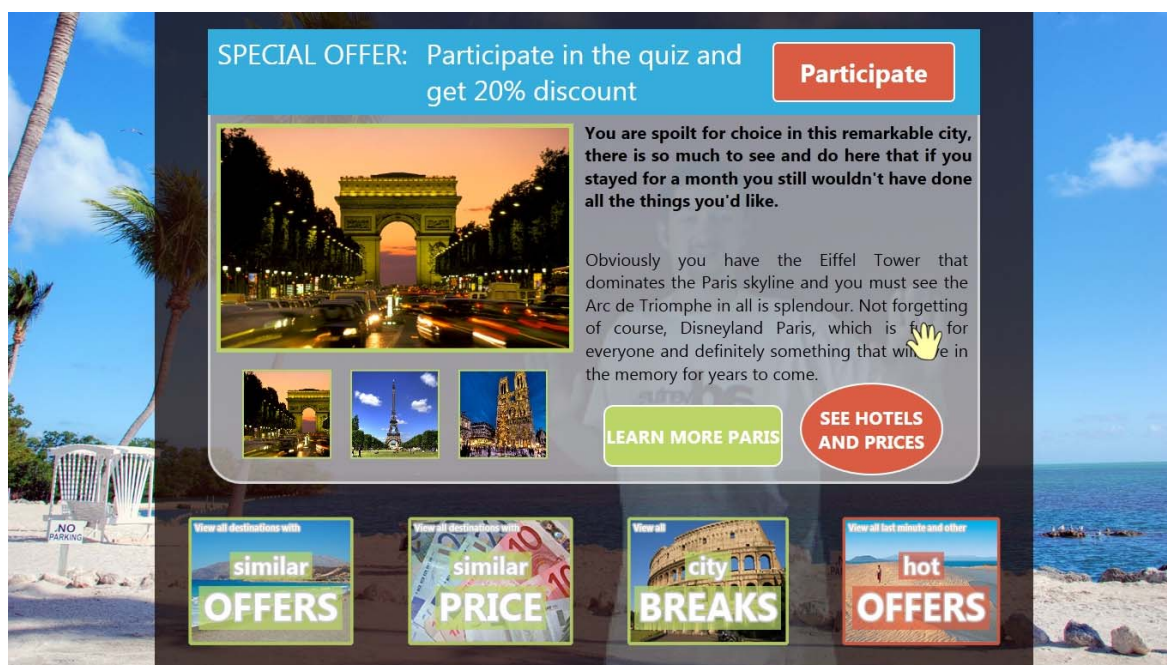
Након тога корисник добија поруку да је могуће да одабере своју жељену дестинацију тако што ће поставити и задржати руку на датој сличици. Или ако жели може гестом померања руке такозваним "swor" да помери карусел који се налази у доњем делу екрана. Ово представља други и трећи задатак учења интеракције.

Поред свега овога најбитније је то што смо корисника већ заинтересовали и привукли му пажњу, ово је нешто ново, нешто што није досадно и нападно. Потрошио је већ пола минута а није ни отпочео интеракцију. Ово се постиже гејмификацијом интеракције. Корисник се буквално игра, учи и истражује. Жели да види шта је то што "ово" нуди.

Након што смо му привукли пажњу време је за креирање додатног интересовања.

Интересовање - Interest

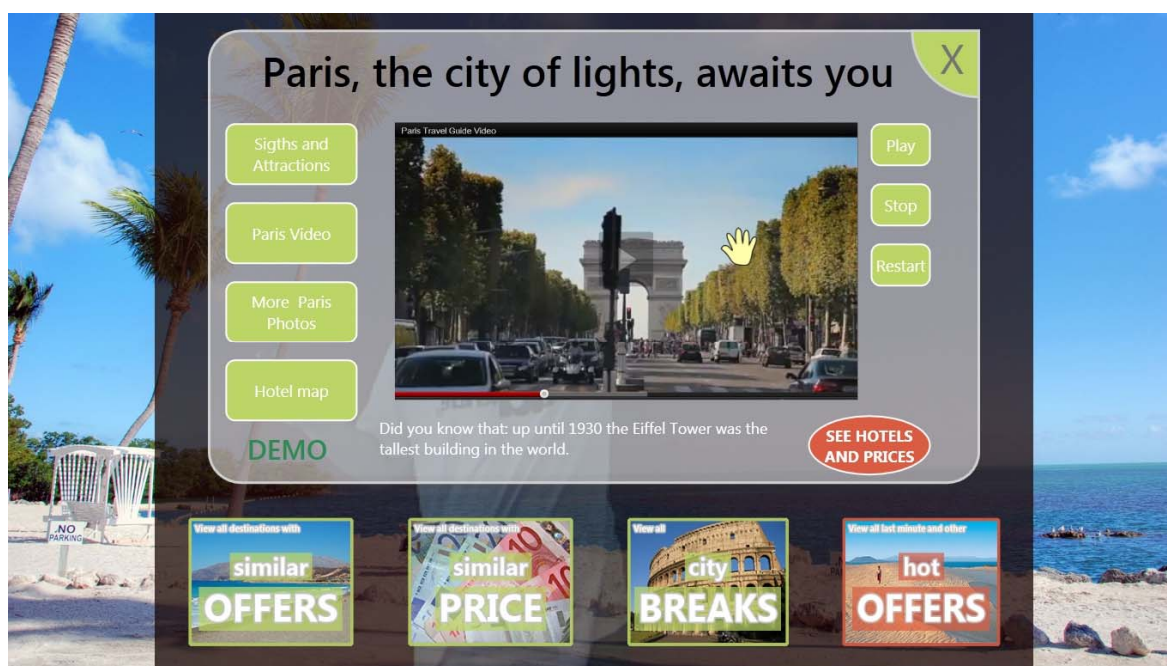
Сада кориснику дозвољавамо комплетну интеракцију са окружењем. Уколико кликне и одабере одређену дестинацију на пример Париз. Добиће контролу која садржи комплетан преглед понуде. Могуће је погледати пар слика Париза, два информативна и мотивациона текста. Такође могуће је отворити нове контроле које нуде прегледање цена и хотела или сазнати више информација о дестинацији.



Слика 3-9 Интеракција корисника са понудама у каруселу

Битно је напоменути корисник у сваком тренутку види себе. Ово је активност која свесно постаје мање битна јер је садржај сада преузео примат у окупирању пажње, али подсвесно корисник има сазнање да он контролише садржај и не осећа се као директни учесник маркетиншке кампање.

Рецимо да корисник одабере опцију "Више информација о Паризу". Добија контролу као на следећој слици. Она нуди вишеслојан садржај. Доступне су мање слике и информације о атрактивностима Париза. Или велика галерија слика добре резолуције. Корисник може погледати видео материјал о дестинацији или погледати Bing мапу са хотелима. Наравно у овој контроли је могуће отићи на ценовник хотела за ову дестинацију.



Слика 3-10 Више информација о Паризу

Као што је могуће видети са Слика 3-10 радна површина је подељена у две целине:

- Позадина - која асоцира на летовање, одмор, генерално лепе ствари и срећне моменте.
- Интерактивну површину где се може видети тренутни видео снимак преко које се приказује садржај. Ова садржај се онда додатно дели на две под целине:
 - Контролну за преглед више информација.
 - Карусел са додатним опцијама.

Додатне опције се референцирају на понуду која се тренутно гледа. Тако да корисник уколико није задовољан овом понудом не мора да крене од почетка и потражи нешто друго, јер ће се у тој ситуацији сигурно окренути и отићи. Наравно постоји могућност да му се понуда допада али да жели да погледа још нешто. Зато се кориснику нуде следеће опције:

- Сличне понуде
- Слична цена

- Градске дестинације - ова опција се мења у зависности од одабране понуде. Јер наравно нећемо кориснику понудити градске излете ако гледа далеке дестинације или дугачка путовања. Примера ради Тајланд или нешто сличног типа. Ово наравно функционише двосмерно.
- Вруће понуде - представља тренутне акције или понуде у последњем минуту.

Рецимо да је корисник одлучио да погледа додатне градске излете. Добиће контролу као на следећој слици.



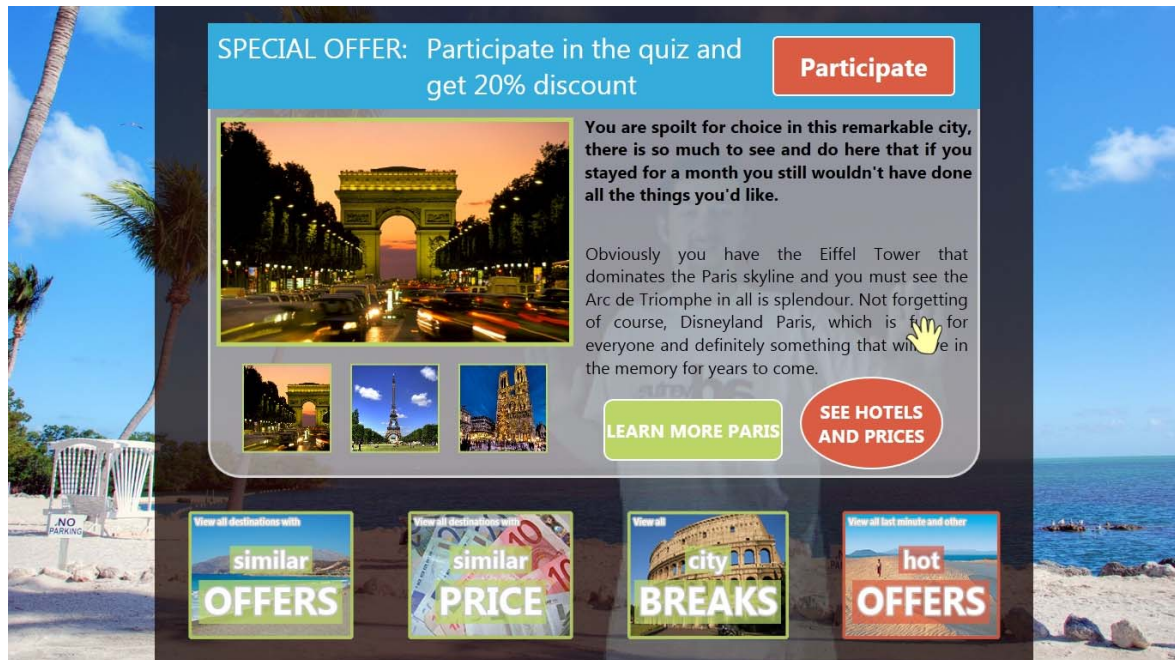
Слика 3-11 Контрола великог карусела

Контролу овде приказану називамо велики карусел, корисник гестом може да се креће лево десно по каталогу понуда. Каталог намерно приказује шест понуда тј. по три у два реда зато што су понуде овако довољно истакнуте али опет се довољно брзо прегледају и проналазе. Довољно је само поставити на руку жељену дестинацију и опет смо на познатом дијалогу за основни преглед понуде. Пошто је корисник ће одабрао и погледао неку понуду покушаћемо да створимо жељу и направимо корак ка продаји.

Стварање жеље за куповином - Desire

Када је корисник отворио одређену понуду након одређеног временског интервала у горњем делу контроле појављује се банер који има задатак повећања ангажовања корисника и створи жељу за куповином. Поруке у банеру су најчешће типа: Учествујте у квизу и освојите 20% попушта на ову дестинацију или било шта сличног карактера, наравно шта ће оба порука бити зависи искључиво од маркетиншка стратегије.

Приказ банера са рекламном поруком је приказан на следећој слици.



Слика 3-12 Банер са рекламном поруком

Када корисник кликне на дугме и жели да учествује у квизу добија крајње једноставну контролу која подсећа на велики број квизова. Ово га чини да се осећа сигурним и као да је на већ познатом терену. Питање треба да буде једноставно и са више понуђених одговора.



Слика 3-13 Интерактивни квиз

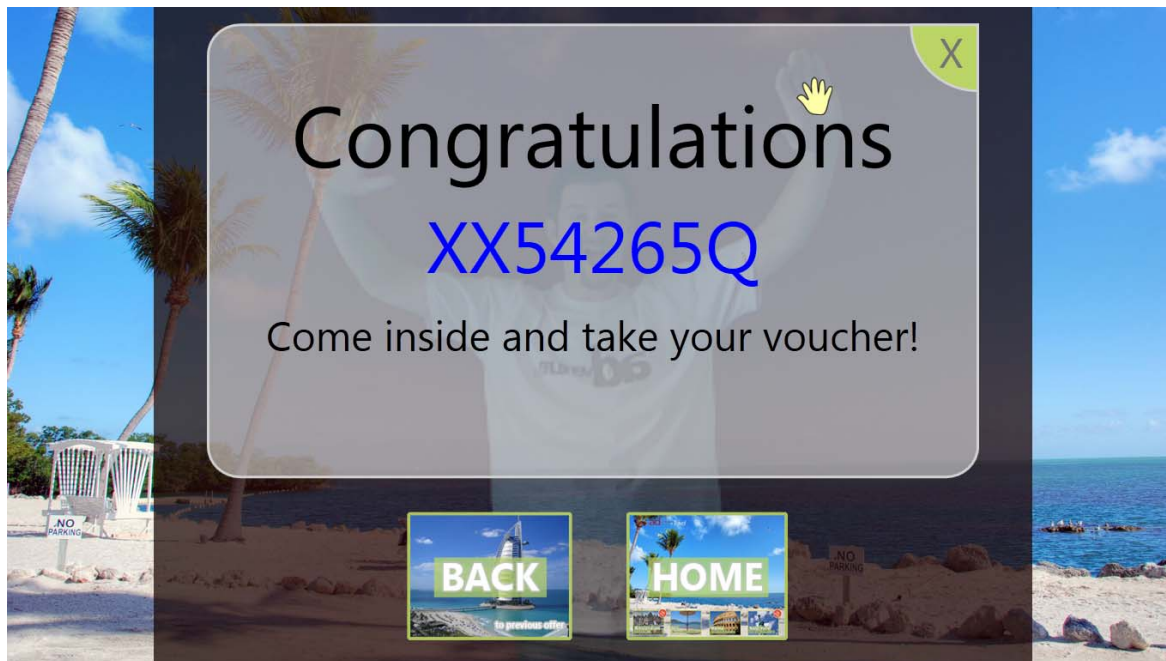
Сама интеракција или стварање жеље код корисника нема никакву сврху ако након овога не дође до акције од стране корисника и не направи корак ка куповини аранжмана. Основна разлика између промоције је то што овде их ви не уводите у агенцију него долазе сами.

Акција - Action

Када корисник одговори на питање врши се провера тачности одговора и уколико није тачно одговорио добија могућност да покуша поново са неким другим питање или да се врати на почетну страницу. Док ако је одговор тачан добија јасну поруку која то истиче и подстиче утисак задовољства. Порука поред себе садржи и код који је потребно прочитати агенту да бу се добио попуст или шта год је већ био циљ квиза.

Корисник то посматра на следећи начин. Одлично добио сам попуст, не могу ништа да изгубим да узмем ваучер, попричаћу са агентом продаје мало и онако сам још нешто хтео да га питам. Имам ваучер не морам сад да купим ништа али добро ми дође за други пут.

А шта се заправо овде дешава. Овде се буквално предност домаћег терена враћа агенцији, корисник долази директно агенту каже да је добио на квизу и жели нешто да пита. Агент користећи своју апликацију може да приступи на основу кода свим подацима шта је дати корисник гледао и да понуди нешто што корисник није видео или да покуша да оствари продају. Заправо овде престаје деловање било које маркетиншке кампање и све се своди на вештину агента продаје. Само што је овде агенту знато лакше јер на основу праћења по подацима он тачно може да претпостави шта корисник воли и шта га занима, зато што је апликација све пратила а модул за анализу обрадио и припремио у виду извештаја за агента.



Слика 3-14 Успешно решен квиз

3.4.1 AIDA

Након што смо се упознали са начином како изгледа и ради интерактиван излога време је да мало попричамо о маркетиншкој теорији која је коришћена приликом креирања апликације. У времену природних корисничких интерфејса апликације никад више неће бити посматране само из информатичког света. Ми ћемо у наставку детаљно анализирати код и поступак како креирања интерфејса тако и контроле покрета и претварање исте у контролисану интеракцију.

Ако погледамо детаљније области по којима је креиран кориснички интерфејс можемо доћи до следећег сазнања:

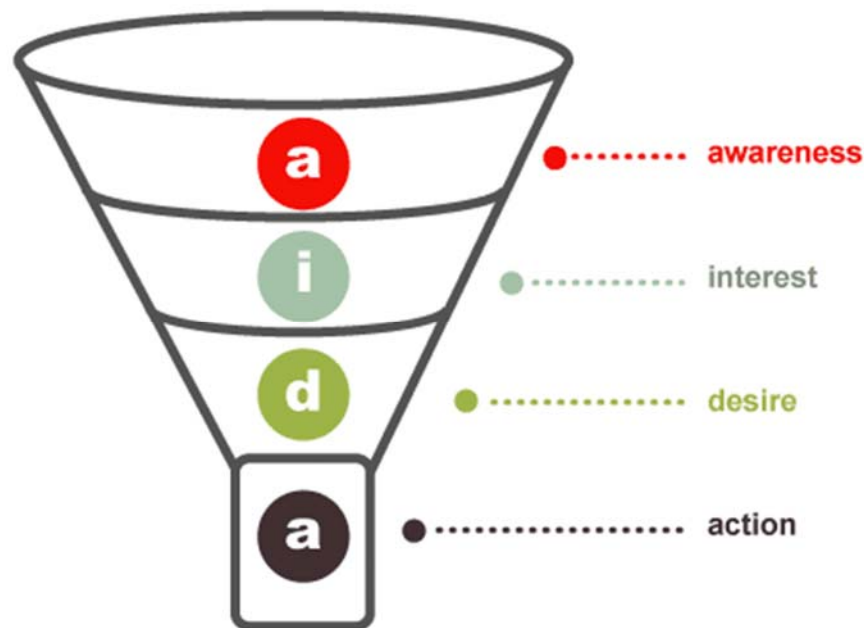
Awareness/Attention - Привлачење пажње

Interest - Интересовање

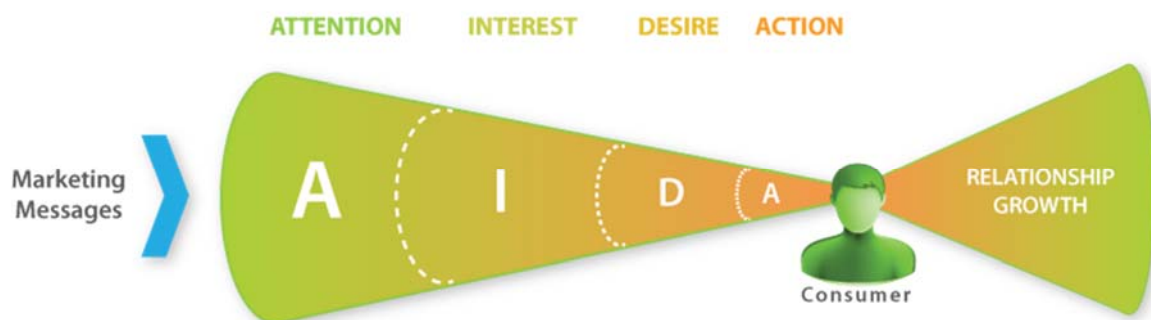
Desire - Стварање жеље за куповином

Action - Акција

AIDA - представља основни маркетиншки појам како заправо корисници би требали да реагују на правилно изведену кампању. Њен основни задатак је да обухвати велику циљну групу која када се "пропусти" кроз филтере омогућава нам добијање јасне групе корисника са којима можемо радити даље и извршити продају. Ово наравно није никаква превара или обмана купаца већ само приступ анализи и припреми успешних кампања.



Слика 3-15 Пример како која група корисника реагује на одређени део маркетиншких порука



Слика 3-16 Утицај порука на корисника

3.4.2 Иницијализација Kinect сензора и употреба у апликацији

Да би Kinect био у потпуности оперативан и да би могли да га користимо у апликацији потребно је инсталирамо SDK. Ово можемо урадити на два начина. Употребити одређени претраживач да пронађемо линк за преузимање најновије верзије а затим исти поступак поновимо и за остале алате. Или можемо једноставно да прикључимо уређај у рачунар и оперативни систем ће преузети све на себе и инсталирати све потребне компоненте у најновијим верзијама. Код овог уређаја назив plug-and-play доноси комплетно ново значење.

Када је поступак инсталације завршене лампица са предње стране уређаја ће трептати да покаже да се сензор спреман и на чекању. У жељени пројекат у форми где желимо да користимо сензор потребно је укупати следећи код. Ово је најбоље урадити одмах након учитавања прозора.

```
// Сада креирамо drawing group коју ћемо користити за цртање
this.drawingGroup = new DrawingGroup();

// Креирање image source који користимо у image control
this.imageSource = new DrawingImage(this.drawingGroup);

// Приказивање видеа
ImageS.Source = this.imageSource;

// Претрага свих сензора и проналажење првог повезаног.
// Ово захтева да је сензор повезан на почетку рада апликације. У другим
// случајевима постоје библиотеке које се за ово користе

foreach (var potentialSensor in KinectSensor.KinectSensors)
{
    if (potentialSensor.Status == KinectStatus.Connected)
    {
        this.sensor = potentialSensor;
        break;
    }
}

if (null != this.sensor)
{
    // Укључивање skeleton stream да добија фрејмове скелета
    this.sensor.SkeletonStream.Enable();

    //Припрема дубинског сензора
    this.sensor.DepthStream.Enable(DepthImageFormat.Resolution640x480Fps30);

    // Укључивање color stream да добија материја са камере
    this.sensor.ColorStream.Enable(ColorImageFormat.RgbResolution640x480Fps30);

    // А лоцирање места за смештање примљених информација о пикселима
    this.colorPixels = new
byte[this.sensor.ColorStream.FramePixelFormat.Length];

    // У овој битмапи ћемо приказати видео на екрану
    this.colorBitmap = new
WriteableBitmap(this.sensor.ColorStream.FrameWidth,
this.sensor.ColorStream.FrameHeight, 96.0, 96.0, PixelFormats.Bgr32, null);

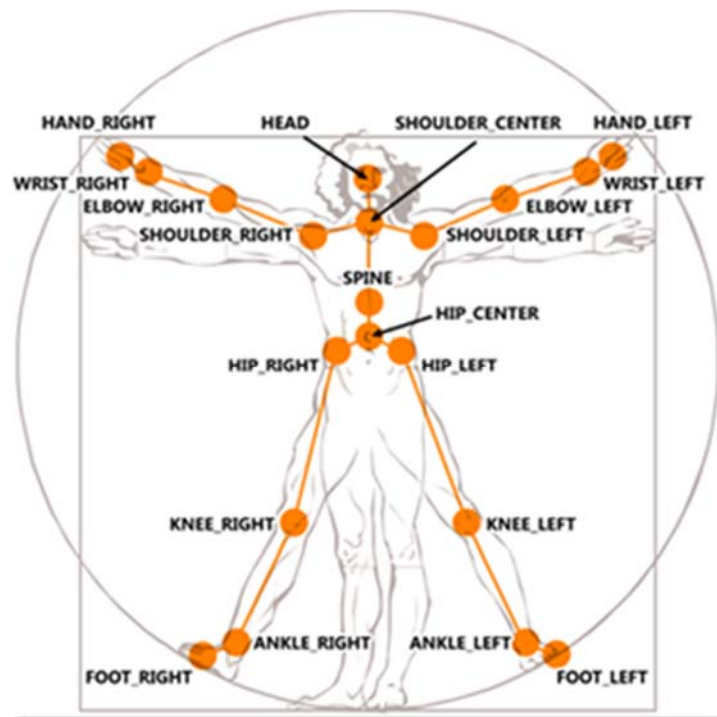
    this.Image.Source = this.colorBitmap;
```

```
// Додавање хендлера за догађаје који ће обрађивати све податке  
добијене са сензора  
  
this.sensor.AllFramesReady += new  
EventHandler<AllFramesReadyEventArgs>(sensor_AllFramesReady);  
  
// Покретање сензора!  
try  
{  
    this.sensor.Start();  
}  
catch (IOException)  
{  
    this.sensor = null;  
}  
}
```

Основна иницијализација сензора је завршена сада је потребно информације које сензор врати обрадити, препознати и управљати.

3.4.3 Управљање покретима

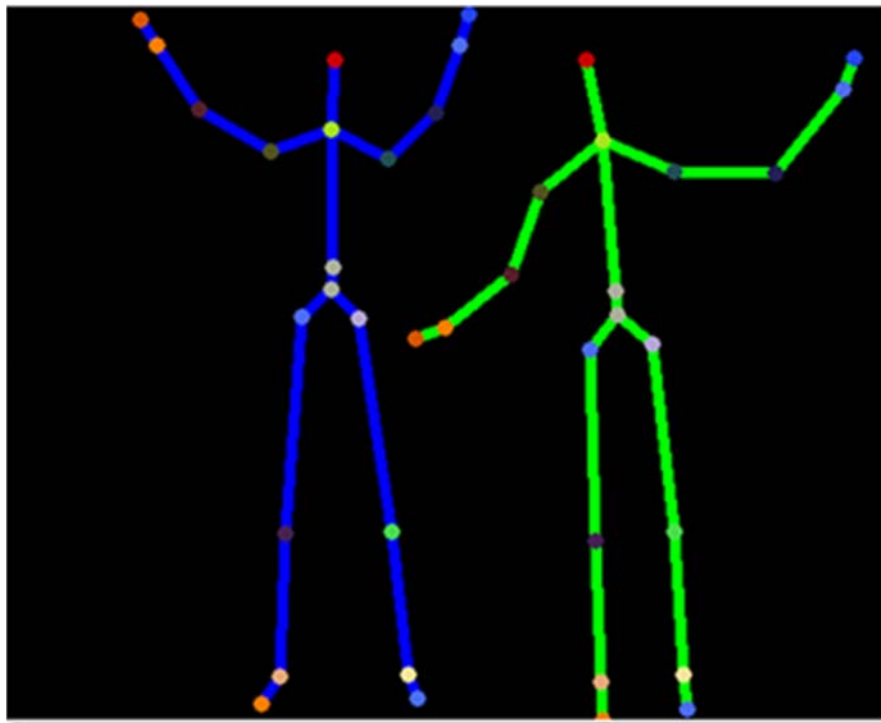
Као што смо раније спомињали овај сензор је у стању да прати двадесет и седам тачака на људском телу. Може да прати шест различитих особа од тога две могу да активно учествују у препознавању покрета. На следећој слици је приказана мапа тачака које се прате.



Слика 3-17 Мапа праћених тачака

Како заправо препознавање покрета ради, кинект је у стању да обрађује укупно тридесет фрејмова/слика у једној секунди. Препознавање са класична камере на овом сензору није ништа другачије у односу на било коју веб камеру и потребно је само тај стрим прихватити и обрадити за употребу.

Што се тиче дубинског сензора он функционише другачије, ласер баца сноп светла који се одбија од објеката, дубинска камера све то снима и пратити сноп светлости који је њој видљив. Софтвер одређује заправо дистанце, као и позицију објекта у тродимензионалном простору. Међутим сав овај посао одрађује драјвер тако да у апликацији не морамо да бринемо о томе. Ако би смо узели пример где имамо два корисника испред сензора и желимо да исцртамо само основне чворове који су међусобно повезани линијама добићемо следећу слику.



Слика 3-18 Приказивање скелета на основни начин

Проблем гужве и великог броја људи смо решили на тај начин што уређај прати све кориснике које може а активан корисник се одређује на један од два начина:

- Најближи корисник се прати а остали проблеми се решавају правилним позиционирањем уређаја у односу на површину на којој се ради. Остали корисници могу да стоје поред или где друго систем ће наставити да прати најближег корисника. Ово је пракса коју примењују најчешће у менијима игрица.
- Постоји минијатурна контрола у једном углу екрана где је потребно да корисник постави руку и онда није више битно да ли је најближи или не систем увек њега прати док се иницијализација управљања не понови.

У примеру кода који ћемо сада показати погледаћемо код који по сваком фрејму обрађује примљене податке и одређује шта ће да ради са њима.

```
using (ColorImageFrame colorFrame = e.OpenColorImageFrame())
{
    if (colorFrame != null)
    {
        // Копирање података о пикселима у привремени низ
        colorFrame.CopyPixelDataTo(this.colorPixels);

        // Уписивање податка у битмапу
        this.colorBitmap.WritePixels(
            new Int32Rect(0, 0, this.colorBitmap.PixelWidth,
this.colorBitmap.PixelHeight),
            this.colorPixels,
            this.colorBitmap.PixelWidth * sizeof(int),
            0);
    }
}

Skeleton[] skeletons = new Skeleton[0];

using (SkeletonFrame skeletonFrame = e.OpenSkeletonFrame())
{
    if (skeletonFrame != null)
    {
        skeletons = new Skeleton[skeletonFrame.SkeletonArrayLength];
        skeletonFrame.CopySkeletonDataTo(skeletons);
    }
    // Ако претпоставимо да нема скелета који су близу
    var newNearestId = -1;
    var nearestDistance2 = double.MaxValue;

    // Претрага кроз све скелете
    foreach (var skeleton in skeletons)
    {
        // Само праћени скелети
        if (skeleton.TrackingState == SkeletonTrackingState.Tracked)
        {
            // Проналажење квадрате раздаљине
            var distance2 = (skeleton.Position.X * skeleton.Position.X) +
                (skeleton.Position.Y * skeleton.Position.Y) +
                (skeleton.Position.Z * skeleton.Position.Z);

            // Да ли је нова дистанца најближа до сад
            if (distance2 < nearestDistance2)
            {
                // Користи нову вредност
                newNearestId = skeleton.TrackingId;
                ActiveSkeletonID = skeleton.TrackingId;
                nearestDistance2 = distance2;
            }
        }
    }
    if (this.nearestId != newNearestId)
    {
        this.nearestId = newNearestId;
    }
    this.activeRecognizer.Recognize(sender, skeletonFrame, skeletons);}
```

Сада када пратимо најближи скелет и имамо све потребне информације потребно је да исцртамо објекат који прати десну руку. Код у наставку је и даље у хендлеру свих стримова.

```
using (DrawingContext dc = this.drawingGroup.Open())
{
    dc.DrawRectangle(Brushes.Black, null, new Rect(0.0, 0.0, RenderWidth,
RenderHeight));

    if (skeletons.Length != 0)
    {
        foreach (Skeleton skel in skeletons)
        {
            if (skel.TrackingState == SkeletonTrackingState.Tracked)
            {
                this.DrawBonesAndJoints(skel, dc);
            }
        }
        // спречава цртање ваз зоне рендеровања
        this.drawingGroup.ClipGeometry = new RectangleGeometry(new Rect(0.0,
0.0, RenderWidth, RenderHeight));
    }
}
```

Затим посматрамо методу **DrawBonesAndJoints**

```
foreach (Joint joint in skeleton.Joints)
{
    Brush drawBrush = null;

    if (joint.TrackingState == JointTrackingState.Tracked)
    {
        drawBrush = this.trackedJointBrush;
    }
    else if (joint.TrackingState == JointTrackingState.Inferred)
    {
        drawBrush = this.inferredJointBrush;
    }
}
```

Слика која представља објекат десне руке може да буде увек присутна па само сакривена на површини док се потреба за њом не јави или је можемо додавати по потреби. У оба случаја неопходно је да водимо рачуна о дубинском индексу видљивости објекта такозваном z index-у.

```
if (drawBrush != null && joint.JointType == JointType.HandRight && ActiveSkeletonID ==
skeleton.TrackingId )
{
    int x = (int)this.SkeletonPointToScreen(joint.Position).X;
    int y = (int)this.SkeletonPointToScreen(joint.Position).Y;
    handRight.Visibility = System.Windows.Visibility.Visible;
}
```

Са сваким померањем корисника врши се ново прерачунавање положаја руке а самим тим и цртање руке. Симулација кретања је на овај начин извршена. Затим је потребно опонашати клик мишем. Потребно је одредити када је потребно симулирати клик мишем. То је најбоље урадити након одређеног временског задржавања руке корисника у жељеној зони. Ако објекат има `MouseClickEvent` подешен сво управљање смо свели на класично Виндоуз програмирање.

Управљање природним корисничким интерфејсом не би било заиста то да не користимо гестове за манипулацију интерфејсом. У овој апликацији смо користили Default Microsoft библиотеку која у себи садржи гестове за скроловање у леву и десну страну. Наравно за ово можемо да користимо и гестове које смо купили, сами из програмирали или креирали у алатима з за прављење гестова. Да доста интеракција је могуће свести на алгоритме колизије али већина савремених интеракција се базира на управљивости гестовима. У нашој апликацији када је гест препознат позивамо методу контроле која извршава промену:

```
private Recognizer CreateRecognizer()
{
    // Инстанцирање препознавања.
    var recognizer = new Recognizer();

    // Препознавање десног геста
    recognizer.SwipeRightDetected += (s, e) =>
    {
        if (e.Skeleton.TrackingId == nearestId)
        {
            if (TestG.Children.Contains(cMoreControl))
            {
                cMoreControl.SwopInvoke(true);
            }
            else
            {
                Carousel.Swop(true);
            }
        }
    };

    // Препознавање левог геста
    recognizer.SwipeLeftDetected += (s, e) =>
    {
        if (e.Skeleton.TrackingId == nearestId)
        {
            if (TestG.Children.Contains(cMoreControl))
            {
                cMoreControl.SwopInvoke(false);
            }
            else
            {
                Carousel.Swop(false);
            }
        }
    };

    return recognizer;}
```

3.4.4 Гејмификација

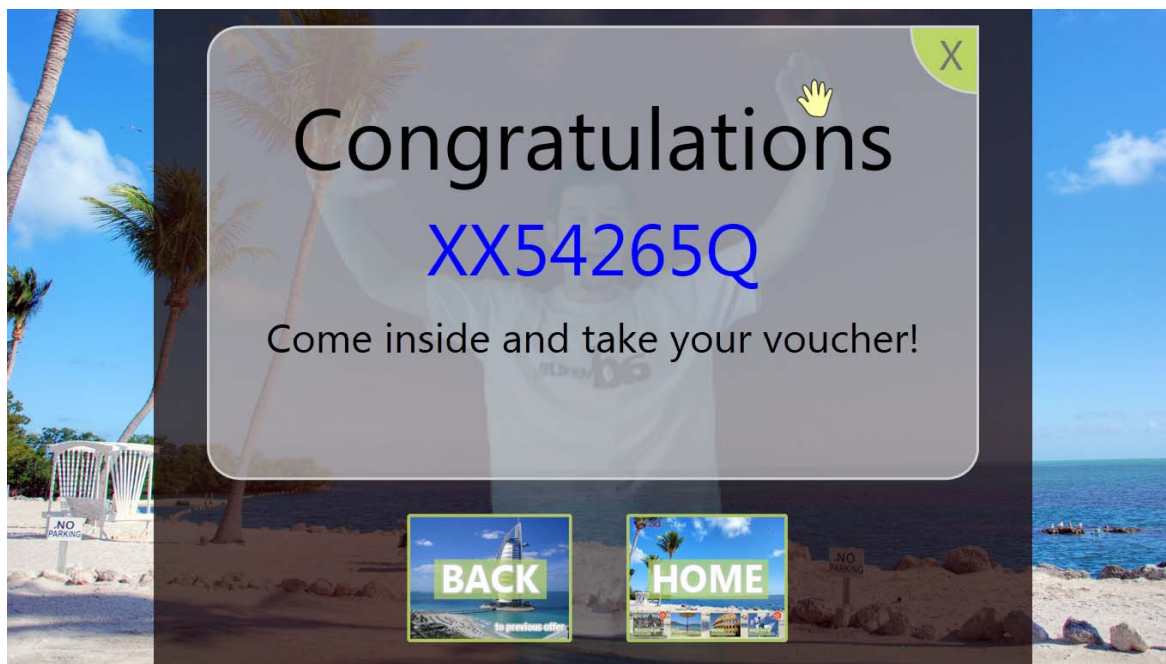
Гејмификација је употреба механике игрица и играчког начина размишљања у контексту који није игрица са задатком да ангажује корисника додатно и помогне у објашњавању или решавању проблема.[8]

Технике гејмификације повећавају људску природну жељу за такмичењем, достигнућима, статусима, само изражавањем и завршецима.

Кључна стратегија за гејмификацију је да обезбеди награде за такмичаре који испуне одређени задатак. Награде могу укључивати бодове, нове нивое или потврде достигнућа. Такмичење је још један елемент игрица који се може користити у гејмификацији. Добијање награде за остваривање задатака који су видљиви другим играчима или постојаност топ листи су добар правац за подстицање других да се такмиче.

Истраживања су показала да гејмификација има велику примену у маркетингу. Магазин Forbes вршио је истраживање над две хиљаде компанија где је укупно 70% компанија одговорило да планира да ове технике у оглашавању и односима задржавања купаца.

У апликацији интерактивног излога коришћена је гејмификација са циљем да додатно ангажује потенцијалног купца да повећа или створи жељу за куповином. Ово је урађено што корисници играју квиз који нуди конкретну награду након успешног решавања. Ово подстиче да се корисник боље осећа након успешног задатка за који добија визуелну потврду која подстиче такмичарски дух код других али ствара и осећај достигнућа.



Слика 3-19 Један од примера гејмификације у апликацији интерактивног излога

Пука употреба ситних игрица у апликацији не би била корисна чак би била и контра продуктивна јер корисник би се онда осећао као особа којој се нешто директно продаје што не би имало повољан исход по агенцију. Зато је кључ у креирању комплетне апликације кроз коју се осећа дух игрице. Потребно је постепено увести корисника у цео нови свет апликације. Овај осећај је

постигнут употребом надграђене стварности. О начину интеракције смо детаљно говорили на почетку овог поглавља док о основама надграђене стварности можете сазнати више из Додатка А. Зато ће фокусу у овом делу бити то како је имплементирана техника праћења корисника и приказивања објеката око главе који знају да прате контуре и не излазе из простора намењеног за цртање без обзира како се корисник креће.

3.4.5 Цртање објеката у надграђеној стварности

Полазимо од основне претпоставке двоосног декартовог координатног система. Он се наравно састоји од четири квадратна. Ако поставимо координатни систем у тачку скелета која представља главу, долазимо до закључка да је потребно да имамо четири положаја објеката који се исцртавају. Пошто технике векторског програмирања дозвољавају креирање вишеслојних објеката нама је потребно да имамо четири различита контејнер објекта који ће у зависности од одбраног елемента то јест понуде имати жељени садржај.



Слика 3-20 Пример једног контејнера попуњеног садржајем

Као што се може видети са претходне слике ово је контејнер за горњи леви квадрант. Када имамо објекат који желимо да прикажемо, имамо почетак координатног система који се освежава тридесет пута у секунди и дистанцу од ивица простора у коме цртамо. Лако је прерачунати када и шта треба да прикажемо.

```
if (drawBrush != null && joint.JointType == JointType.Head && ActiveSkeletonID ==  
skeleton.TrackingId)  
{  
    tempPeth.Source = new BitmapImage(new  
Uri("/SopotDemoApp;component/Images/Callouts/01"+ActiveCity1+".png", UriKind.RelativeOr  
Absolute));  
    double CallOutX = (this.Width -  
(this.SkeletonPointToScreen(joint.Position).X * scale.ScaleX));  
    double CallOutY = (this.Height -  
(this.SkeletonPointToScreen(joint.Position).Y * scale.ScaleY));  
    MainCanvas.Visibility = System.Windows.Visibility.Visible;  
    tempPeth.Width = 300;  
    tempPeth.Height = 220;  
    double sirinaSlike = tempPeth.Width / scale.ScaleX;  
    double visinaSlike = tempPeth.Height / scale.ScaleY;  
  
    if (this.SkeletonPointToScreen(joint.Position).X > sirinaSlike &&  
this.SkeletonPointToScreen(joint.Position).Y > visinaSlike)
```

```

{

    MainCanvas.Children.Clear();
    MainCanvas.Children.Add(tempPeth);

    Canvas.SetLeft(tempPeth,
this.SkeletonPointToScreen(joint.Position).X * scale.ScaleX - tempPeth.Width-50);
    Canvas.SetTop(tempPeth,
this.SkeletonPointToScreen(joint.Position).Y * scale.ScaleY - tempPeth.Height);
}
else if (this.SkeletonPointToScreen(joint.Position).X >
sirinaSlike && this.SkeletonPointToScreen(joint.Position).Y < visinaSlike)
{

    tempPeth.Source = new BitmapImage(new
Uri("/SopotDemoApp;component/Images/Callouts/02" + ActiveCity1 + ".png",
UriKind.RelativeOrAbsolute));
    tempPeth.Height = 220;
    tempPeth.Width = 300;

    MainCanvas.Children.Clear();
    MainCanvas.Children.Add(tempPeth);

    Canvas.SetLeft(tempPeth,
this.SkeletonPointToScreen(joint.Position).X * scale.ScaleX - tempPeth.Width - 50);
    Canvas.SetTop(tempPeth,
this.SkeletonPointToScreen(joint.Position).Y * scale.ScaleY);
}
else
{
    MainCanvas.Children.Clear();
}
if (this.SkeletonPointToScreen(joint.Position).Y > visinaSlike &&
(1024 - this.SkeletonPointToScreen(joint.Position).X * scale.ScaleX) > sirinaSlike *
scale.ScaleX)
{

    tempPethDL.Source = new BitmapImage(new
Uri("/SopotDemoApp;component/Images/Callouts/03" + ActiveCity2 + ".png",
UriKind.RelativeOrAbsolute));
    tempPethDL.Height = 220;
    tempPethDL.Width = 300;
    if (!(tempPethDL.Parent == MainCanvas))
    {
        MainCanvasRight.Children.Clear();
        MainCanvasRight.Children.Add(tempPethDL);
    }
    Canvas.SetLeft(tempPethDL,
this.SkeletonPointToScreen(joint.Position).X * scale.ScaleX+50 );
    Canvas.SetTop(tempPethDL,
this.SkeletonPointToScreen(joint.Position).Y * scale.ScaleY - tempPethDL.Height);
}
else if (this.SkeletonPointToScreen(joint.Position).Y <
visinaSlike && (1024 - this.SkeletonPointToScreen(joint.Position).X*scale.ScaleX) >
sirinaSlike*scale.ScaleX)
{

```

```
        tempPethDL.Source = new BitmapImage(new
Uri("/SopotDemoApp;component/Images/Callouts/04" + ActiveCity2 + ".png",
UriKind.RelativeOrAbsolute));
        tempPethDL.Height = 220;
        tempPethDL.Width = 300;

        MainCanvasRight.Children.Clear();
        MainCanvasRight.Children.Add(tempPethDL);

        Canvas.SetLeft(tempPethDL,
this.SkeletonPointToScreen(joint.Position).X * scale.ScaleX+50);
        Canvas.SetTop(tempPethDL,
this.SkeletonPointToScreen(joint.Position).Y * scale.ScaleY);
    }
    else
    {
        MainCanvasRight.Children.Clear();
    }
}
```

Поступак исцртавања објеката као што можете видети је идентичан цртању десне руке корисника, једина разлика је у позиционирању слика у односу на корисника.

3.4.6 Праћење подацима (Data tracking)

Праћење подацима је нови појам који смо извели на основу техника праћења корисника на интернет порталима, у суштини примењује се нова технологија у потпуно нову сврху али са основним концептом као код ових техника. Ове технике се зову Behavioral targeting.

Behavioral targeting се односи на широки опсег технологија и техника коришћених од стране интернет оглашивача које им омогућавају повећање ефикасности кампање тако што прикупљају податке генерисане од стране посећених сајтова или страница [9]. У интернет свету се може чак и сматрати и нарушавањем права приватности, јер то што претражујете можда не желите да поделите са било ким. У нашој апликацији овог проблема нема, јер излог је јавна површина, оно што прегледате је у сваком тренутку видљиво свима а систем само бележи податке о посетама. (Напреднија верзија бројања саобраћаја).

Како функционише Behavioral targeting, корисник посети одрешену интернет страницу, страница која је посећена, време проведено на свакој страници, линкови на које је кликнуо, претраге које врши или ствари са којима врши интеракцију, дозвољавају сајту да скупља податке или друге факторе на основу којих се креира профил који се повезује се корисниковим прегледачем.

Као резултат власници сајта могу да употребе податке да креирају сегменте прилагођене одређеном типу публике тј. корисницима сличних профила. Када се корисник врати на одређени сајт или мрежу сајтова користећи исти интернет прегледач, ови профили се користе да оглашивачи знају који тип рекламе да прикажу. Ово доноси већи број прегледа реклама али и могућност одређивања веће цене за одређене типове реклама за које компанија која их оглашава може да тврди да ће бити боље приказане.



Слика 3-21 Како интернет сајтови прате понашање својих корисника [10]

Behavioral маркетинг се може користити у конјункцији са осталим типовима таргетирања приликом оглашавања као што су географско, демографско или контекстуално. Овај процес се често и назива тражењем циљне групе.

Праћење подацима у нашој апликацији је само једна од статистичких метода које се користе за успех анализе кампање. Систем као што смо већ спомињали скупља информације о великом броју догађаја који се дешавају са интеракцијом корисника и апликације. Са једне стране све промене које се прате омогућавају генерално бољи увид у ефикасност кампање или пословања агенције. Увек и у сваком послу је боље знати шта од производа иде а шта не, шта је то што народ жели и на који начин.

Праћење подацима омогућава агентима продаје да када разговарају са корисником знају шта је корисник гледао од понуда, колико се времена задржавао и на којој категорији, који је то опсег цена до које може да иде итд.

Да агенти на овај начин стичу предност, јер имају информације шта понудити кориснику али резултат овакве претраге је у томе што ће корисник мање времена потрошити да објасни шта жели или на који начин жели. Такође није свако добар у изражавању својих жеља и захтева.

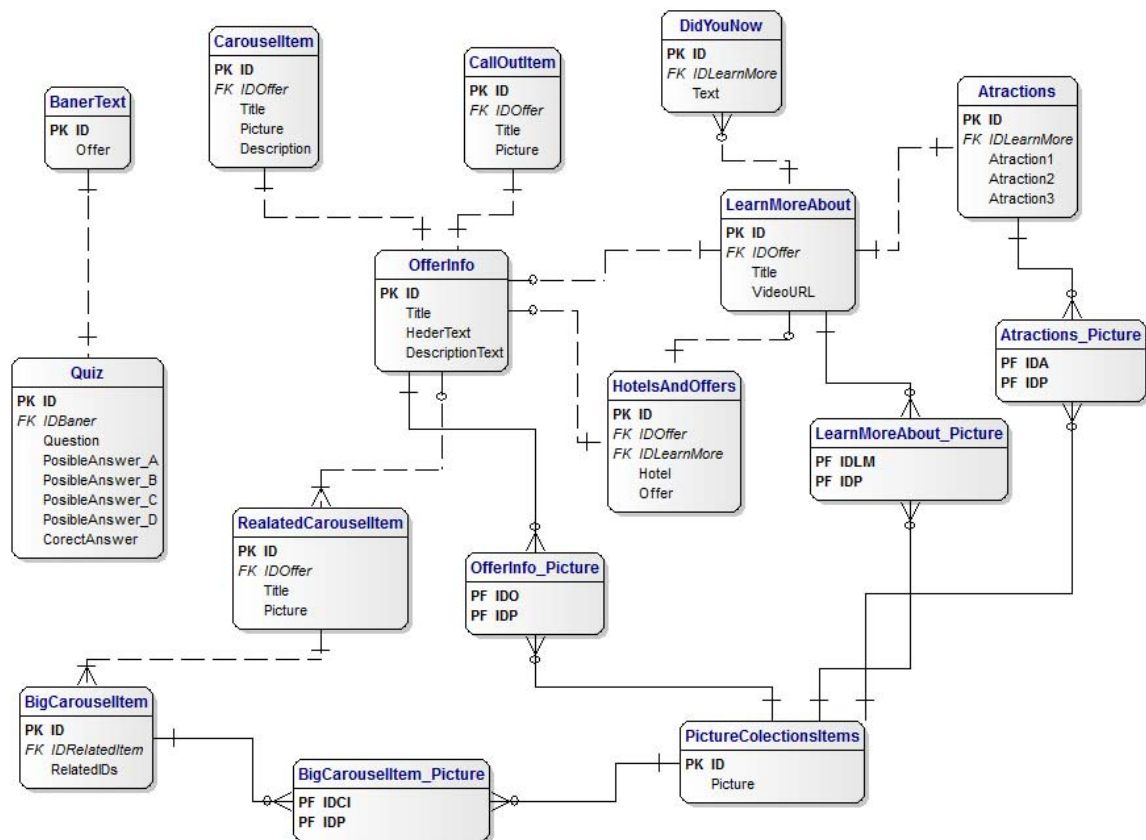
На овај начин се пружа личнија и услуга ближа људима, приснија. Да не спомињемо то колико је овај начин пословања продуктивнији. Када је функционисање малих и средњих агенција у питању квалитет пружања услуге и степен продуктивности су од круцијалног значаја за опстанак.

3.5 Централни сервер

Централни сервер представља срж обраде и складиштења података комплетног информационог система. На њему је инсталиран релациони систем управљања базама података - SQL Server. Такође сервер је подешен да може да ради и као мали Web сервер и на себи има Internet Information Service (IIS) који хостује Web сервисе који имају задатак комуникације посредовања између базе података и клијената за прикупљање података у апликацијама.

3.5.1 База података

Пошто смо се већ детаљно упознали са појединачним корисничким апликацијама потребно је да размотримо структуру централне базе која чува податке неопходне за функционисање система. На следећој слици приказана је шема базе података:



Слика 3-22 Шема базе података

OfferInfo - Кључна табела што се из апликација могло и закључити је табела са понудама, око ње је ово складиште података орјентисано.

	Atribut	Tip podatka
PK	ID	Int
FK	Title	Varchar

	HeaderText	Varchar
	Description Text	Text

Свака понуда је смештена у ову табелу. Понуда је дефинисана својим кључем, има наслов и два текста који је описују. Референцира се на следеће табеле:

- CarouselItem - Свака ставка карусела је референцирана на једну понуду на коју води директно. Релација је односа 1:1.
- CallOutItem - Свака објекат AP је референциран на једну понуду на коју води директно. Релација је односа 1:1.
- RelatedCarouselItem - Свака понуда има неколико адекватних група сличних понуда. Сличне понуде се приказују у главном каруселу за време прегледа понуде. Релација је односа 1: N.
- HotelAndOffers - Свака понуда има одређени број хотела повезаних са њом. Релација је односа 1: N.
- LearnMoreAbout - О свакој понуди је могуће погледати опширније информације. Релација је односа 1:1.
- OfferInfor_Picture - За једну понуду је могуће везати колекцију слика од три слике. Пошто је релација односа m:n. Ова табела представља међу релацију као помоћ приликом референцирања.

Напомена:

Табеле које представљају међу релације услед односа m:n. Неће бити додатно описане осим као референце на основне табеле.

Табела **PictureCollectionItems** - велики број контрола у апликацијама захтева произвољан број слика у колекцијама које приказује. Потребно је било обезбеди сигуран начин складиштења слика и њиховог чувања и проналажења. Пошто број слика може превазићи могућности успешног архивирања у локалним директоријумима а опет са друге стране то нису велике количине за систем базе података. Одлучено је да се ови подаци чувају у оквиру базе података.

	Atribut	Tip podatka
PK	ID	Int
	Picture	Image

Табеле које користе колекције слика су:

- BigCarouselItem
- OfferItem
- LearnMoreAbout
- Attractions

Табела **LearnMoreAbout** - Ова табела представља проширење базне табеле о понудама. Њен основни задатак је да чува све информације које додатно описују неку понуду.

	Atribut	Tip podatka
PK	ID	Int
FK	IDOffer	Int
	Title	Varchar
	VideoURL	Varchar

Табела **LearnMoreAbout** се референцира на следеће табеле:

- **DidYouKnow** - Сваки шири опис неке понуде садржи неколико занимљивих мање познатих чињеница о датој понуди. Релација је односа 1: N.
- **Attractions** - Додани опис сваке понуде садржи неколико привлачних дестинација које треба посетити. Једна колекција од укупно три дестинације је везана за један проширен поглед на понуду.
- **HotelsAndOffers** - Свака понуда има одређени број хотела повезаних са њом. Релација је односа 1: N. Хотелима је могуће приступи из основне или проширене понуде.
- **LearnMoreAbout_Picture** - За једну понуду је могуће везати колекцију више слика. Пошто је релација односа m:n. Ова табела представља међу релацију као помоћ приликом референцирања.

Табела **HotelsAndOffers** има задатак да чува све информације о понудама хотела релевантне за дати аранжман. Хотелима се може приступити и из општих информација и из контроле за више информација.

	Atribut	Tip podatka
PK	ID	Int
FK	IDOffer	Int
FK	IDLearnMore	Int
	Hotel	Varchar
	Offer	Text

Табела **Attractions** - Представља проширење табеле за детаљнији опис понуде. Основни задатак јој је да чува податке о најпожељнијим дестинацијама за посету у оквиру дате понуде. Сваки објекат атракције са собом садржи и колекцију слика.

	Atribut	Tip podatka
PK	ID	Int
FK	IDLearnMore	Int
	Attraction1	Text
	Attraction2	Text
	Attraction3	Text

Ова табела се референцира на следећу табелу:

- **Attractions_Picture** - Објекат атракције садржи колекцију слика где је свака слика везана за једну атракцију. Могуће је имати максимално три атракције по понуди.

Табела **DidYouKnow** - Сваки шири опис неке понуде садржи неколико занимљивих мање познатих чињеница о датој понуди. Ова поља нису обавезна за додатни опис понуде у апликацији.

	Atribut	Tip podatka
PK	ID	Int
FK	IDLearnMore	Int
	Text	Text

Табела **CallOutItem** - Објекти који се користе за надграђену стварност приликом кретања корисника смештени су у овој табели. Сваки објекат води на једну одређену понуду. Табела је састављена на следећи начин:

	Atribut	Tip podatka
PK	ID	Int
FK	IDOffer	Int
	Title	Varchar
	Picture	Image

Табела **CarouselItem** - Сви објекти који се користе у главном каруселу као и објекти који се користе у анимацији када корисник није присутан су смештени у овој табели. Један објекат одговара једној понуди. Табела је састављена на следећи начин:

	Atribut	Tip podatka
PK	ID	Int
FK	IDOffer	Int
	Title	Varchar
	Picture	Image
	Description	Varchar

Табела **RealtedCarouselItem** - Сваку понуду је могуће сврстати у неки виртуелну категорију. Да ли су то јефтинија путовања, летовања, зимовања или пак одређене дестинације. Самим тим свака понуда може имати групу сличних понуда. Ова табела има задатак да чува елементе који се

приказују у главном каруселу када је нека понуда већ отворена. И на основу одабране ставке да води до нове велике контроле за каруселни приказ дестинација.

	Atribut	Tip podatka
PK	ID	Int
FK	IDOffer	Int
	Title	Varchar
	Picture	Image

Ова табела се референцира на следећу табелу:

- BigCarousellItem - Сваки објекат овог типа може имати са собом повезано једну или више страница објеката великог карусела. Самим тим релација је односа 1: N.

Табела **BigCarousellItem** - Ова табела чува објекте који се приказују у великом каруселу приликом одабира неке од повезаних опција.

	Atribut	Tip podatka
PK	ID	Int
FK	IDRelatedItem	Int
	RelatedIDs	XML

Ова табела се референцира на следећу табелу:

- BigCarousellItem_Picture - Представља колекцију слика које се користе приликом попуњавања великог карусела.

Табела **BanerText** - Приликом приказивања главне понуде банер који се појављује попуњен је садржајем из ове табеле. Табела је веома једноставна и садржи само поруке које се приказују.

	Atribut	Tip podatka
PK	ID	Int
	BanerText	Text

Ова табела се референцира на табелу Quiz у релационом односу 1:1.

Табела Quiz садржи податке за интерактивни квиз који се приказује у апликацији. Њен садржај представљају основни подаци квиза као што су : Питање, потенцијални одговори (укупно четири) и тачан одговор. Табела је састављена на следећи начин:

	Atribut	Tip podatka
PK	ID	Int
FK	IDBaner	Int
	Question	Text
	PosibleAnswer_A	Varchar
	PosibleAnswer_B	Varchar
	PosibleAnswer_C	Varchar
	PosibleAnswer_D	Varchar
	CorectAnswer	Int

3.5.2 Сервис за комуникацију са базом података

Сервис за комуникацију са базом података представља Web сервис који има основни задатак да премости сву комуникацију апликација са базом података преко себе и на тај начин обезбеди додатну дозу стабилности информационог система као и унифицирани начин приступа подацима. Поред тога може да одреди приоритетне активности у виду синхроних и асинхроних позива итд.

Сервис се може користити за три одвојена процеса:

- Обезбеђивање података апликацији интерактивног излога.
- Прерачунавање и обезбеђивање статистичких података за агентску апликацију.
- Попуну и одржавање централне базе података.

Методе које припадају сервису поделићемо према претходно дефинисаним групама. Прва група има задатак похрањивања информација превасходно намењених излогу. Наравно свака од ових метода може се користити где год је то потребно:

- **GetOffers()** - Ова метода враћа колекцију објеката Offer који представља изузетно сложен објекат. Међу његовим својствима могуће је наћи колекције Хотела и аранжмана као и слика локације у понуди. Такође овај објекат у себи садржи и објекат детаљнијег описа понуде који у својим својствима садржи следеће колекције:
 - атракције
 - слике
 - "да ли сте знали" информације.
- **GetCarouselItems()** - Враћа колекцију свих објеката који могу бити приказани у главном каруселу или могу бити коришћени и почетној анимацији.
- **GetCallOutItems()** - Враћа колекцију објеката који се користе у надграђеној стварности.
- **FindPicture(int ID)** - Уколико је потребно може вратити само једну слику. Параметар методе је референтни идентификациони број.
- **GetAllRelatedItems()** - Враћа колекцију објеката који се користе у главном каруселу када је нека од понуда одабрана.
- **RelatedItem(int IDOffer)** - Уколико је то потребно може да врати једну одређену ставку карусела.

- **GetBigCarouselItems(int IDRelatedItem)** - За сваку ставку која је приказана у главном каруселу када је одабрана нека понуда може да врати ставке великог карусела у зависности од одабране ставке.
- **FindHotel(int ID)** - Уколико је то потребно може вратити информације о само једном одређеном хотелу.
- **GetBanerText()** - Враћа колекцију објеката које се приказују у банеру и референцирају на квиз.
- **GetQuizItems()** - Враћа све објекте који се користе у квизу.
- **FindQuiz(int IDBaner)** - Уколико је потребно може пронаћи одређено питање из квиза на основу познатог банера.

Статистичке информације као што је већ претходно описано представљају доста битан део овог вида комуникација са корисником и начина оглашавања. Заправо видели смо да се маркетиншке стратегије формирају на основу ових информација.

- **AddUser()** - Када систем детектује новог корисника ту информацију прослеђује систему за анализу кроз ову методу.
- **AddEvent(int EventTypeID , int UserID)** - Било каква акције корисника која је релевантна за статистику се уписује овом методом. Параметри методе представљају редни број догађаја у енумерацији статистички битних догађаја., други параметар је идентификациони број корисника.
- **CalculateStatistics()** - Метода које се извршава пре повлачења статистичких информација, покреће поступак анализе и припрема податке за извештаје.
- **CalculateStatisticsAsync()** - Ова метода ради исто што и претходна али се њено извршавање обавља асинхроно.
- **UserLives()** - Одлазак корисника се детектује и позива се ова метода. Апликација се након тога ресетује.
- **GetVisitedOffers()** - Враћа статистичке информације везано искључиво за погледане понуде.
- **GetVisitTimes()** - Враћа информације о времену задржавања корисника.
- **GetDataByUser(int QRCode)** - За одређеног корисника на основу кода враћа све статистичке информације.
- **GetTimeOfVisitsData()** - Враћа податке везане за период дана када се десила интеракција везана за дане и месеце.
- **GetPeopleData()** - Враћа све податке посматрано из угла посета по кориснику.
- **GetOverallData()** - Сва све информације неопходне за креирање великог извештаја.
- **ClameQRCode(int UserID)** - Када корисника реши успешно квиз, њему се додељује код.

На крају потребно је споменути методе које служе за одржавање садржаја базе података. Ове методе раде операције додавања то су методе које почињу са Add или операције брисања и почињу са Delete. Због изузетне сличности метода овде ће бити описане само методе додавања. Методе брисања се могу посматрати кроз исту призму.

Из назива метода могуће је практично закључити њихову намену али хајде да погледамо шта која метода ради:

- **AddOffer()** - Додавање нове понуде, потребно је испунити доста захтеваних услова да би све зависности биле задовољене пре додавања у базу података ово ради администраторска апликација.
- **AddCarouselItem()** - Додавање нове ставке карусела.
- **AddHotel()** - Додавање новог хотела.
- **AddCallOutItem()** - Додавање новог елемента у надграђену стварност.
- **AddRelatedItem()** - Додавање новог повезаног елемента са понудом.
- **AddBannerTextItem()** - Додавање новог банер текста.
- **AddQuizQuestion()** - Додавање новог питања за квиз.
- **AddBigCarouselPage()** - Додавање нове странице великог карусела.

Свака од ових метода у себи садржи истоимени објекат класе која се уписује у базу али то је изостављено због поједностављења приказа.

3.6 Проблеми приликом реализације и експлоатације

Иако је у претходном делу овог поглавља доста речи било о правилној изради и употреби апликације која користи сензоре покрета постоје одређени проблеми или потенцијални проблеми који могу направити проблем у свакодневnoj експлоатацији. У овом делу поред проблема говорићемо и о могућим ризицима.

Прво поште је овде реч о информационом систему, морамо посматрати све могуће проблеме који се могу десити у експлоатацији. Стабилан напон струје је свакако неопходан, нестанци и прекиди могу направити проблем електронским уређајима. Зато би требало предложити уградњу UPS уређаја у систем испред сваког сервера. Конекција би искључиво требала да се одвија преко локалне кабловске мреже а не бежичним путем. Сервери који имају излаз на интернет не би требали да буду директно повезани на интернет уређаје већ преко рутера. Позиционирање уређаја за презентовање је исто веома битан фактор. Уколико је то реч о излогу великих проблема не би требало да буде, али ако се користе у чекаоницама, сајмовима или било који другим прометним местима, мора се водити рачуна о позиционирању каблова као и месту пролажења људи. Ово морамо размотрити јер уређаји за праћење су конфигурисани на одређену дистанцу, уколико особе пролазе између корисника и система, процес препознавања и праћења биће прекинут.

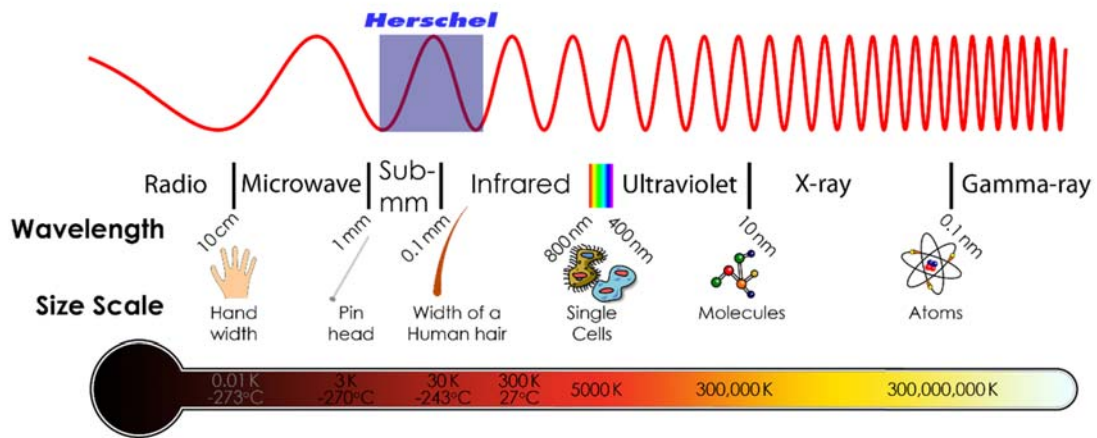
Такође ово нису уређаји који су намењени употреби ван канцеларија, јер прашина и киша могу наштети сензорима.

Поред свих ових ризика инфрацрвени сензори тренутно имају један велики недостатак. То је утицај директне сунчеве светлости на препознавање објеката за чију се детекцију користе. Заправо уколико се користе у нормалним пословним просторијама, сајмовима или тржним центрима никаквих проблема нема. Чак коришћење и на отвореним површинама ноћу није проблем ако се остали ризици искључе.

Овде до проблема долази услед спектар сунчеве светлости и снаге сунчевих зрака. Заправо ово није генерални проблем са инфрацрвеним уређајима већ само са онима који су у продаји за препознавање покрета данас. Што значи да ако имамо продавницу са излогом у пешачкој зони интерактивни излог без додатних модификација неће радити.

Наравно овај проблем је могуће решити тако што ће се купити додатни емитер инфрацрвене светлости који је знато јачи, чији ће сноп зрака бити видљив дубинској камери. Да би се ново

формирани спектар видео на сензору потребно је монтирати и нови филтер који пропушта баш одговарајући опсег са којим камера може да ради. У овом случају реч је о 980 nm филтерима и емитерима.



Слика 3-23 Спектар светлости и величина зрака

Уколико су сви ризици покривени апликација је спремна за употребу у свим могућим условима. Наравно успех апликације онда ће зависити од позиционирања корисничког интерфејса према кориснику или од квалитета садржаја.

#

4 НАЧИН И УПОТРЕБА

У овом поглављу упознаћемо се основним начинима инсталације и пуштања у рад интерактивног излога. Видећемо које основне хардверске и софтверске услове систем мора да задовољава као и шта је од иницијалних тестова потребно обавити пре почетка употребе.

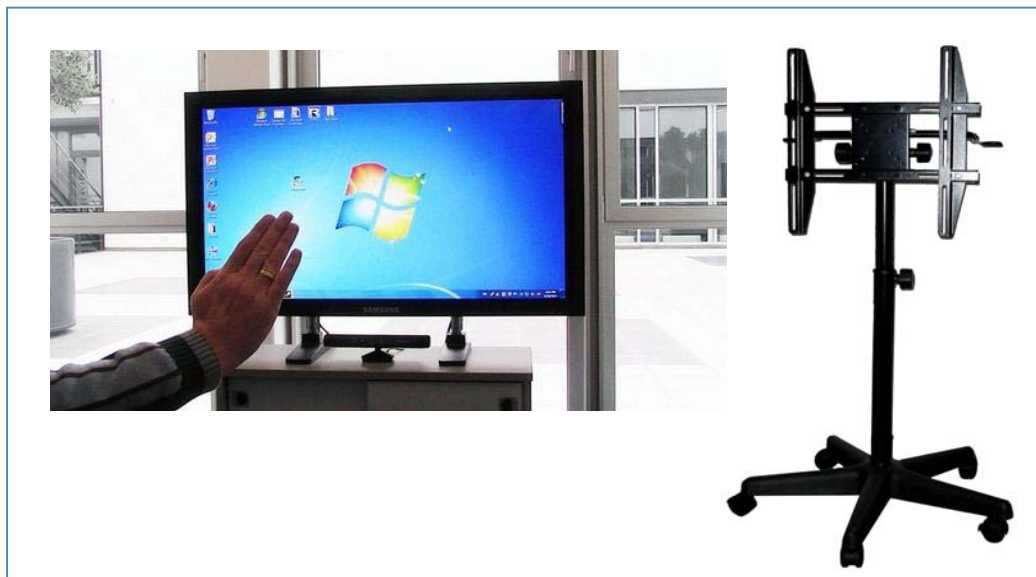
Такође погледаћемо методом корак по корак употребу апликације или основне смернице за употребу апликације или одржавање централне базе податка.

4.1 Инсталација интерактивног излога

Излог је замишљен да пројектује слику велике резолуције на раздаљини од пола метра у односу на стакло излога са унутрашње стране. Као што смо већ спомињали ово је могуће извести телевизорима велике дијагонале или пројекторима са кратком дистанцом емитовања. Одабрали смо прву опцију због једноставнијег померања целе инсталације као и мање цене крајњег производа.

За инсталацију излога као што нам је познато из поглавља 3.2 потребне су нам следеће компоненте:

- LCD TV - Потребно је да буде дијагонале од 50 инча, пуне HD резолуције или другачије речено од 1920 x 1080p. Потребно је да има стандардне VGA и HDMI портове.
- Microsoft Kinect за Windows
- Сервер за покретање апликације излога следеће или јаче конфигурације :
 - Процесор - Intel® Pentium® Dual-Core G860, BOX 32nm
 - Меморија - 6GB
 - Графичка картица - VGA AMD Radeon 7850, 1GB/DDR5/HDMI/DVI/256bi
 - Капацитет хард диска - 250 GB



Слика 4-1 Пример хардверских компоненти потребних за инсталацију излога

- Централни сервер следеће или јаче конфигурације -
 - Процесор - CPU LGA1155 Intel® Core™ i5-3350P, 3.10GHz/6MB BOX 22nm

- Меморија - 8GB
 - Графичка картица - VGA AMD Radeon 7850, 1GB/DDR5/HDMI/DVI/256bi
 - Капацитет хард диска - 1 TB
- Носач TV-а - пожељено је да носач буде са точкићима и променљиве висине да би могао да одговара различитим условима експлоатације.
 - Мрежни каблови - Реч је о класичним мрежним кабловима намењеним за рад у затвореним површинама.
 - VGA кабл - веза између сервера за излог и дисплеја се обавља преко класичног аналогног кабла, било које дигиталне каблове је могуће користи али због мале иницијалне резолуције сензора за праћење покрета, никакав додатан квалитет се од њих не може добити. Овај кабл би требало да буде дужине од 3 до 5 метара.

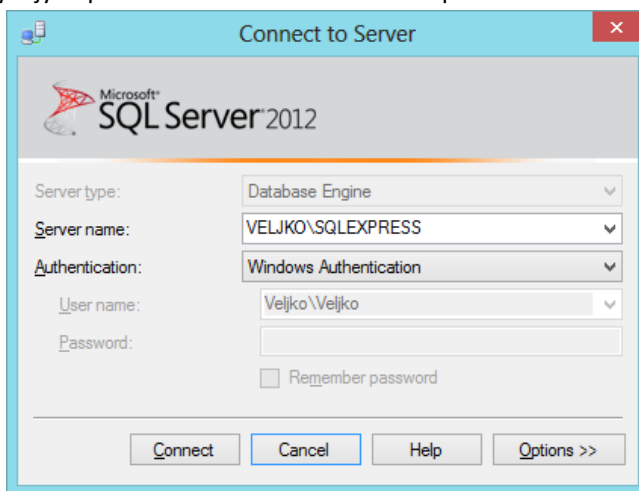
Када су сви хардверски захтеви задовољени потребно је погледати које софтверске захтеве је потребно испунити.

Централни сервер је конфигурисан следећим алатима:

- Оперативни систем - Windows Server 2008 / 2012
- Потребно је инсталирати све верзије .NET framework од верзија 2.0 до 4.5 . Верзије 2.0 и 3.5 су нам потребе због Веб сервиса а 4.0 због апликације. Потребно је наравно проверити шта је инсталирао у старту са оперативним системом, велики део ових пакета јесте саставни део оперативног система.
- SQL Server 2008 R2 или SQL Server 2012
- Internet Information Service - IIS

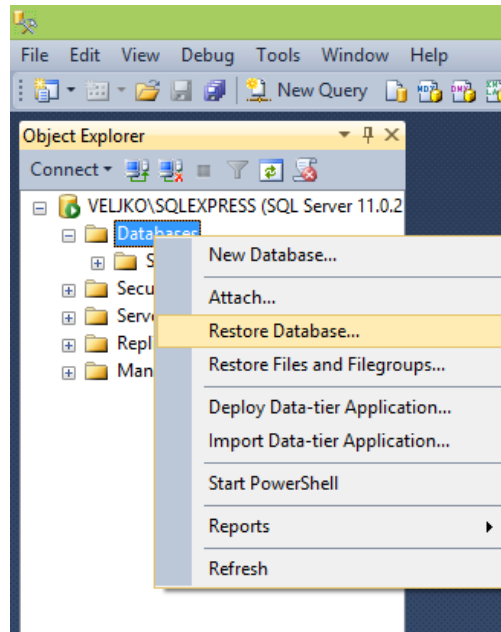
Када су све компоненте повезане и инсталиран је сав потребан софтвер, потребно је поставити централну базу података. Пре свега потребно је конфигурисати све корисничке налоге и права приступа и одлучити се за начин приступа серверу. Велики број апликација које раде са базама података приступају преко SQL корисника а не преко класичних Windows корисничких налога. За процес додавање базе најбоље је користи Microsoft SQL Server Management Studio. Поступак постављања, било које постојеће, базе преузете са неког другог сервера је следећи:

1. Повезивање на сервер - Потребно је унети назив сервера и подесити права приступа. Ми смо у овом случају користили основни налог из оперативног система.



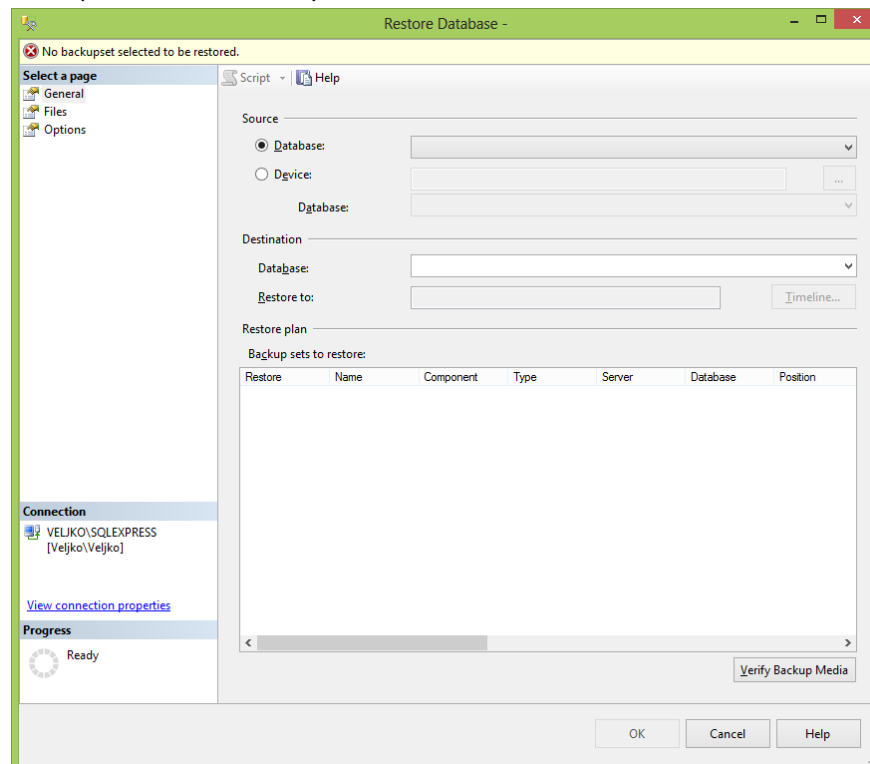
Слика 4-2 Повезивање на сервер базе података преко стандардног налога

- Кликните десним тастером миша на директоријум **Databases** у **Object Explorer** менију. Одаберите опцију **Restore Database**.



Слика 4-3 Постављање базе 2 корак

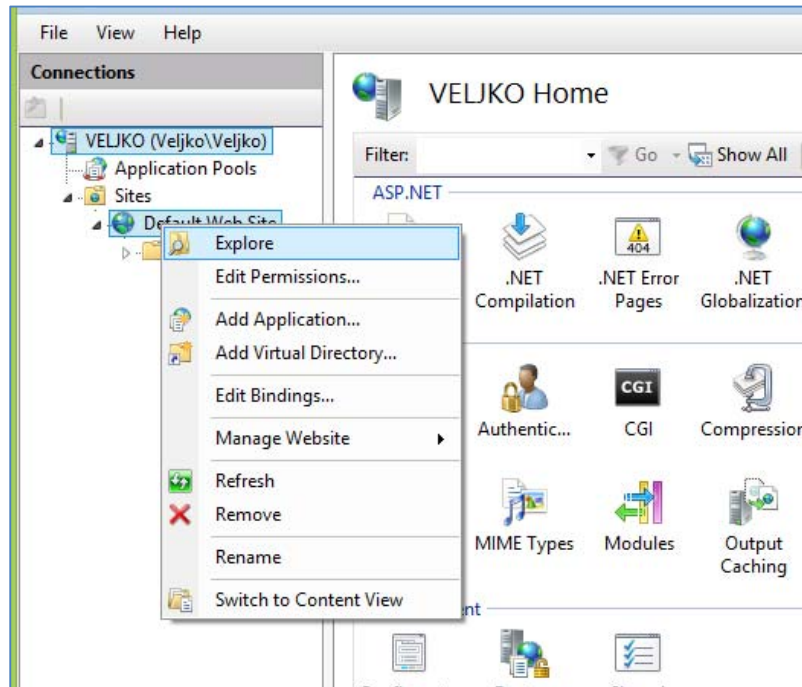
- Потребно је попунити назив базе података и одабрати фајл из кога се постављање базе података врши. За назив базе унесите **ISWCentralDB** и кликните на **OK**.



Слика 4-4 Постављање базе корак 3

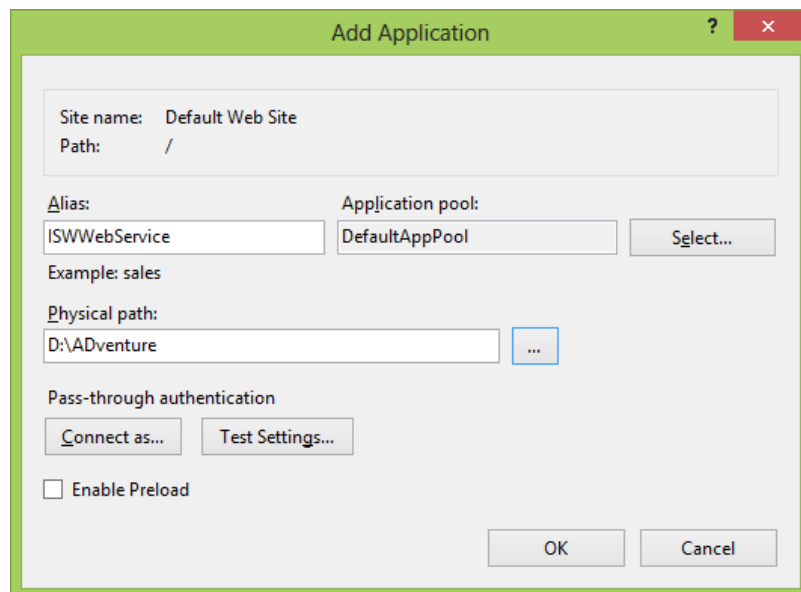
Након постављања базе података потребно подесити сервис који комуницира са базом. Поступак постављања веб сервиса је једноставан и потребно поновити следећу процедуру:

1. Из контролног панела оперативног система одаберите опцију **Internet Information Service**.
2. Кликните десним тастером миша на **Default Web Site** и одаберите опцију **Add Application**.



Слика 4-5 Додавање новог веб сервиса корак 1

3. У прозору за додавање нове апликације потребно је попунити следеће параметре: **Alias - ISWebService**, за путању одаберите путању до директоријума где је смештен сервиса и уколико је потребно (што у овом случају није) подесите **Application pool** на жељену верзију .NET framework-a.



Слика 4-6 Додавање новог веб сервиса корак 2

Сервер излога је конфигурисан следећим алатима:

- Оперативни систем: Windows 7 или Windows 8
- .NET framework 3.5 и 4.0 у зависности од оперативног система инсталиран је један, други је потребно инсталирати ручно.
- Инсталирати Интерактиван излог као класичну Windows installer апликацију.

Кориснички рачунари на којима се извршава Агент апликација морају да имају инсталирани .NET framework 4.0 и да буду у локалној мрежи рачунара.

4.1.1 Конфигурација мрежних параметара и иницијални тестови

Приликом постављања сервера у локалну мрежу требало би водити рачуна приликом конфигурисања адреса. Уколико су претходно познате слободне адресе онда нема проблема јер сервис и апликације се могу променити пре припреме инсталације да раде са датим адреса у супротном конфигурација адреса је следећа.

- 192.168.200.1 - адреса централног сервера
- 192.168.200.2 - адреса сервера излога
- Од 192.168.200.1 до 192.168.200.10 су резервисане адресе од стране ИС.

Адресе рачунара на којима се извршава агент апликација уколико нису у заузетом опсегу се могу користити слободно.

Да би смо могли да потврдимо да систем је задовољио све хардверске, софтверске и мрежне услове, потребно је да обавимо низ иницијалних тестова:

- Са сервера излога пинговати централни сервер.
- Из интернет прегледача на серверу излога покренути веб сервис.
- Стартовати тест методу која провера повезаност сервера са базом.
- Покренути апликацију интерактивног излога.
- Тестирати понашање сензора у дневним условима.
- Подесити позицију сензора у односу на експлоатационе услове.
- Пробати апликацију према упутству приложеном у следећем одељку.

Уколико је су сви тестови успешно завршени излог је спреман за рад, уколико неки од тестова није прошао, поновите цео поступак администрације а ако и даље систем не функционише контактирајте администратора.

За проверу исправног рада агент апликације потребно је урадити следеће:

- Проверите да ли је апликација правилно инсталирана на циљаном рачунару.
- Проверити да ли су присутне све неопходне софтверске компоненте.
- Стартујте апликацију.
- Креирајте општи извештај.

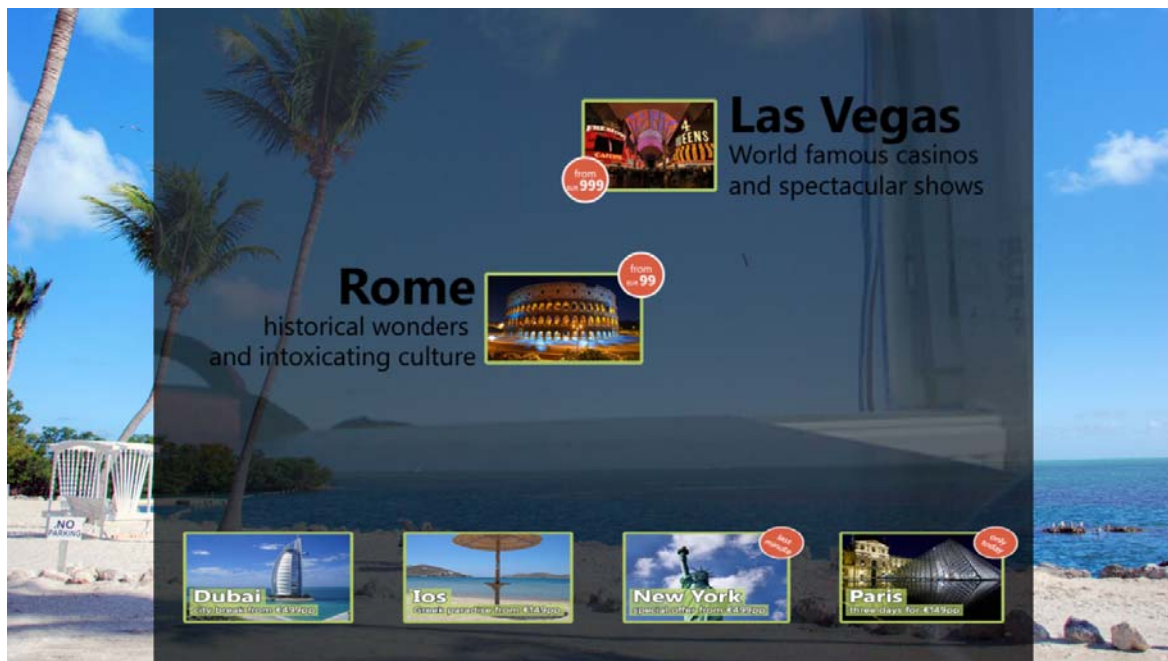
4.2 Упутство за употребу FrontPanel апликације

Апликацију интерактивног излога можемо још и назвати FrontPanel апликацијом, јасно је из којих разлога. Да би смо били сигурни да је апликација спремна за несметану и константну употребу потребно је пратити упутство сачињено од следећих ставки:

- Покретање апликације и уводна анимација.
- Интеракција корисника са апликацијом преко надграђене стварности и праћења покрета.
- Отварање основне понуде.
- Прегледање више информација о понуди.
- Играње квиза

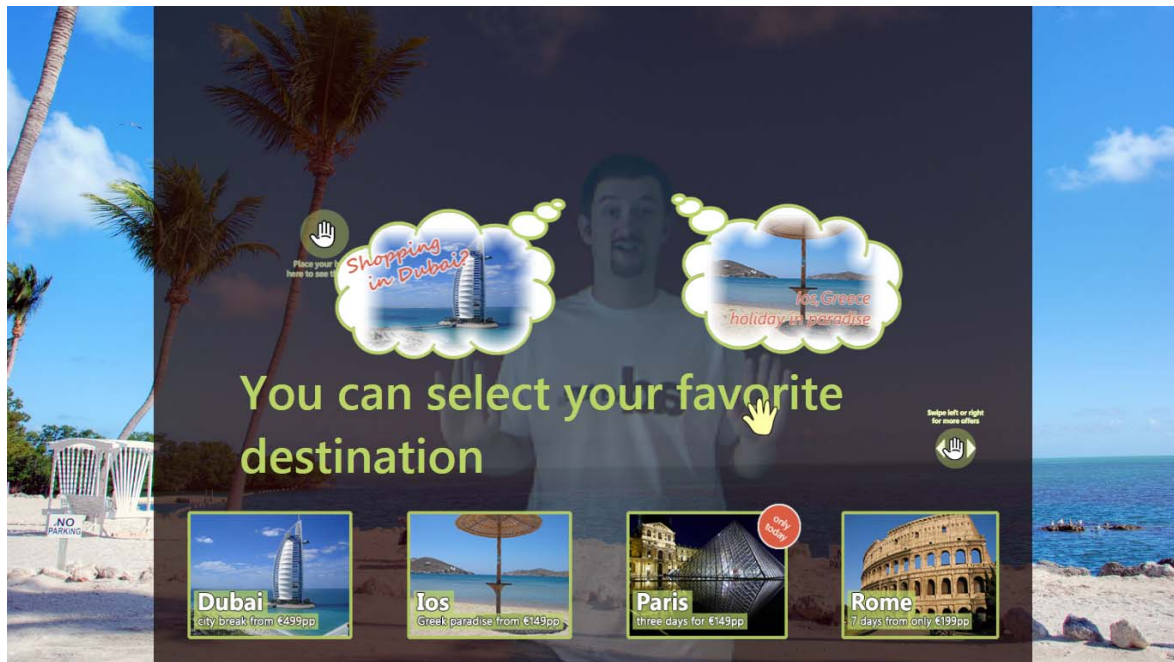
Пошто смо се сагледали све битне ставке функционисања FrontPanel апликације у наставку ћемо по принципу корак по корак проћи комплетну употребу апликације:

1. Физичким покретањем сервера након инсталација апликације, подешен је одређени број команди које се извршавају приликом стартовања рачунара. Једна од ново додатих командни је за аутоматско покретање апликације приликом стартовања рачунара. Уколико је апликација добро инсталирана видећете следећу уводну анимацију:



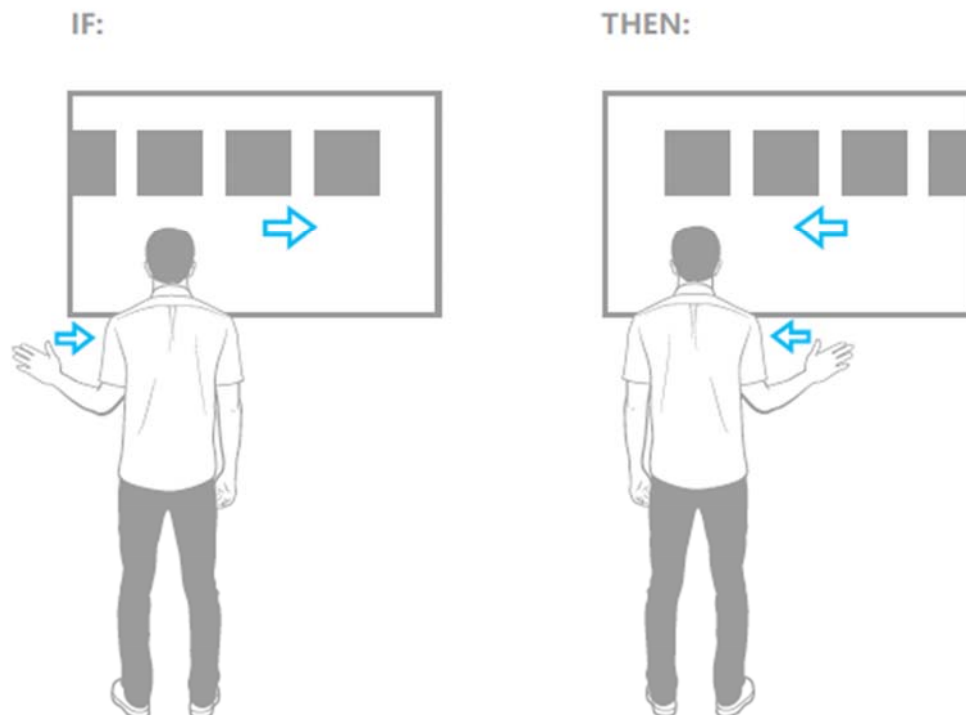
Слика 4-7 Уводна анимација

2. Закорачите испред сензора, апликација ће вас препознати, аутоматски прекинути анимацију и можете видети себе на радно површини.
3. Уколико се померите лево или десно облачићи који се исцртавају поред ваше главе Вас прате.
4. Након 20 секунди уколико нисте отварали ни једну понуду добијате поруку како је могуће манипулисати радним окружењем.

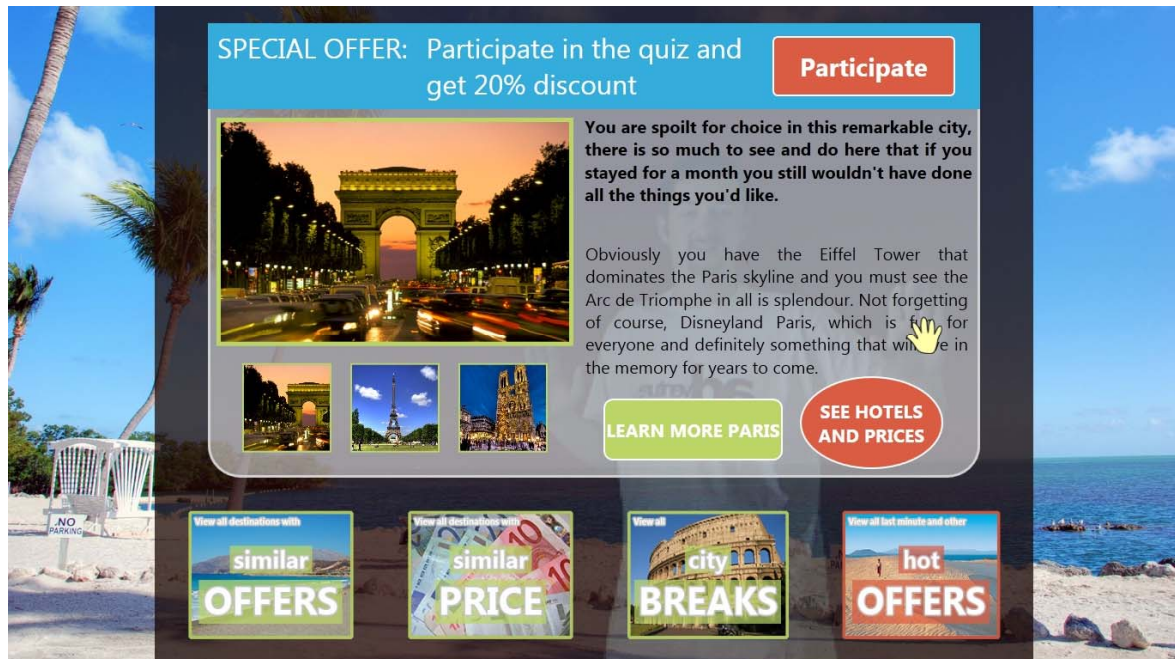


Слика 4-8 Изглед поруке након 20 секунди не активности

5. Према упутству са следеће слике извршите гест померања руке, овај гест се назива "swipe" и омогућава манипулацију главним каруселом у апликацији.

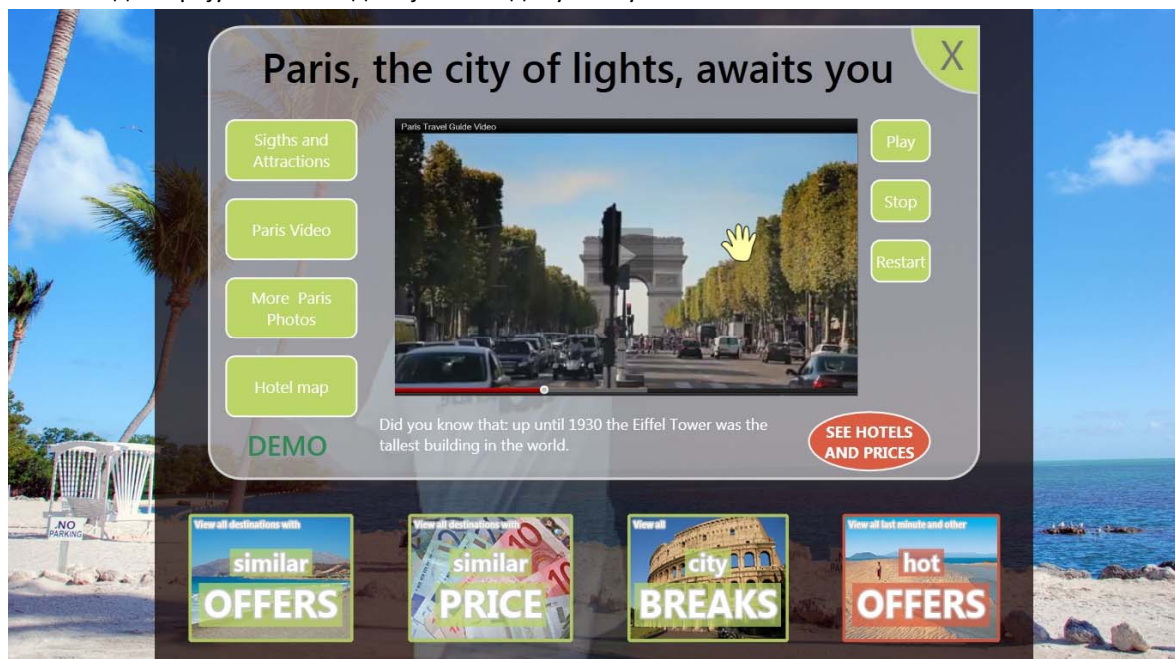


6. Померајте десну руку и приметите како нацртана рука на екрану прати позицију ваше руке.
7. Поставите руку на иконицу за Париз. Сачекајте тренутак да отворите понуду о Паризу.



Слика 4-9 Прегледање понуде за Париз

8. Пређите руком преко слика, задржите пар тренутака и прегледајте доступне слике.
9. Да би смо могли да погледамо више информација о Паризу потребно је поставити руку на одговарајуће поље и добијамо следећу слику.



Слика 4-10 Прегледање додатних информација о Паризу

10. Одаберите опцију Sights and Attractions
11. Одаберите опцију Видео, покрените , паузирајте и рестартујте видео. (Напомена, видео је само доступан у ситуацији да су услови конекције ка интернету испуњени.)

12. Одаберите опцију More Paris Photos, и прегледајте фотографије доступне за дестинацију. Да би сте погледали фотографију, довољно је да пређете риком преко ње.
13. На крају одаберите опцију за преглед мапа хотела.

Сада смо се упознали са основним опција у начинима манипулације апликацијом. Уколико желите можете одабрати опцију City Breaks из главног карусела и добиће те следећу слику.



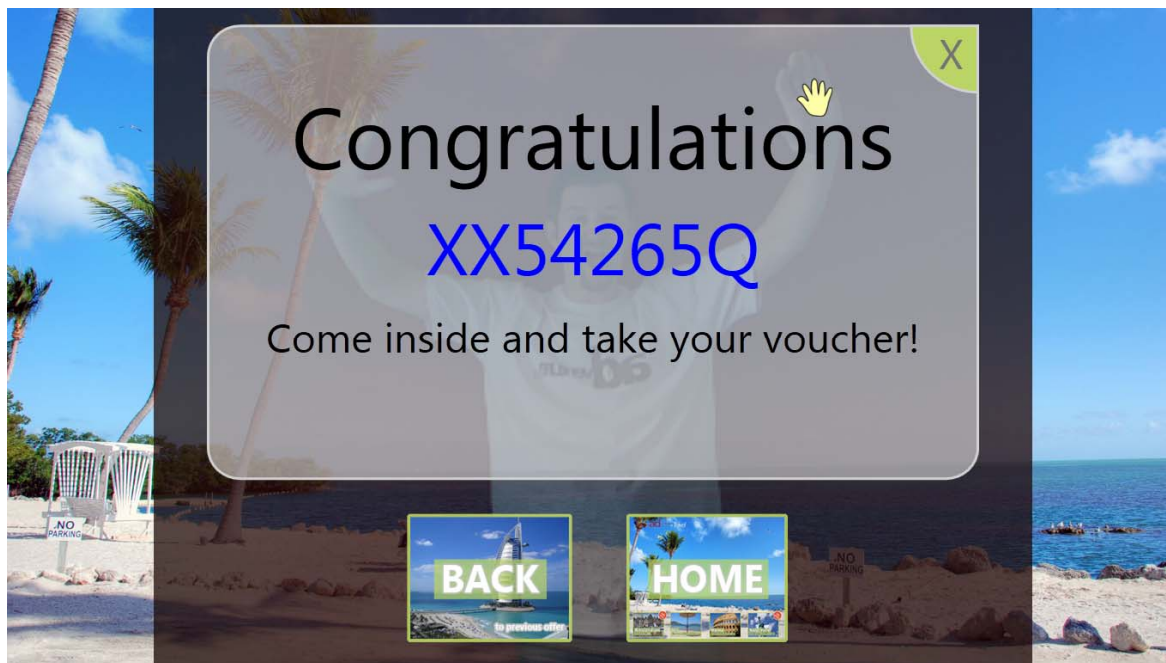
Слика 4-11 Велики карусел

14. Поновите swiре гест и поново пронађите Париз међу дестинацијама. Добијате екран као на слици 4.9. Уколико је понуда отворена дуже од пет секунди у горњем делу екрана појављује се порука која позива на играње квиза. Поставите руку на опцију Participate.



Слика 4-12 Покретање квиза

15. Појављује се квиз као на слици 4.12. Одговорите на питање и сачекајте одговор. Уколико сте тачно одговорили добиће те следећу слику.



Слика 4-13 Завршетак квиза

16. Кликните на картицу за повратак на почетак. Апликација је сада спремна за експлоатацију. Након изласка корисника из видокруга сензора апликација се сама ресетује на почетно стање.

4.3 Одржавање централног сервера

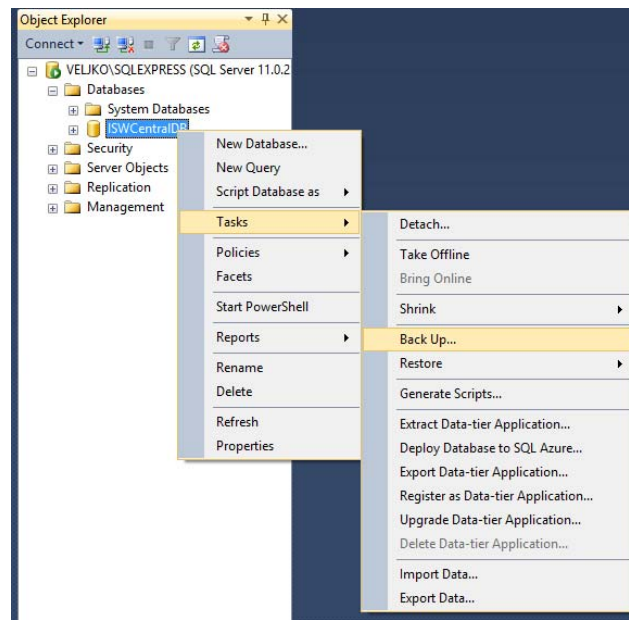
Одржавање централног сервера представља једну од битних ставку за исправно функционисање система. Централна база података се константно попуњава догађајима битним за статистичке анализе. Такође садржи све икада увезене податке у њу. Зато је од круцијалног значаја вршити редовно чување података снимањем копија базе, такозваних бекап-ова.

Предложен начин администрирања сервера је следећи:

- Једном месечно урадити бекап базе.
- Пре било које измене у бази или ради освежавања верзија ИС или других промена на серверу обавезно је урадити бекап.
- Све промене које захтевају рестартовање сервера потребно је радити у времену када није велики промет испред излога. Пожељно ноћу.
- Покретање system update-а је потребно радити у истим терминима као и промене из претходне ставке.

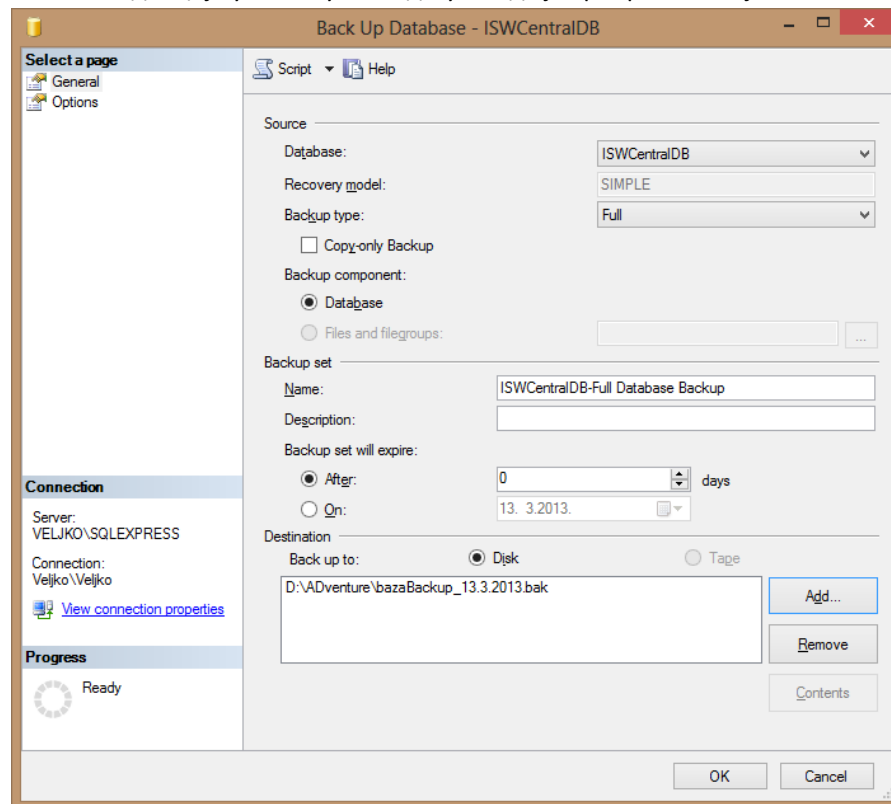
У следећим корацима описан је поступак креирања бекапа централне базе:

1. Покрените алат за одржавање сервера базе података.
2. Конектујте се на сервер.
3. Из менија Databases пронађите ISWCentralDB и кликните десним тастером на базу.
4. Из менија одаберите Task → Back up... .



Слика 4-14 Покретање опције за креирање бекапа базе података

5. Сада је потребно одабрати где и под којим именом желимо да сачувамо копију базе.
6. Кликните на пронађите жељену локацију и унесите име фајла. Неписано правило је да се због боље евиденције у назив упише датум када је креирана копија базе.



Слика 4-15 Креирање копије базе

7. Када су сви параметри подешени кликните на ОК . Завршен је процес копирања базе.

5 БУДУЋНОСТ ПРИМЕНЕ И РАЗВОЈА

Ако посматрамо пример интерактивног излога и апликације које се користе као огласна решења са сигурношћу можемо тврдити да ће се корак ка масовној употреби ових технологија десити! Можемо очекивати да на свих већим и прометним местима имамо следеће генерације интерактивних апликација које приказују поруке оглашивача, као да ће цене ових технологија у будућности додатно пасти. Заправо ако посматрамо различите петогодишње визије великих информационих компанија стиче се један заједнички утисак а то је да ће технологије интерактивних стакала доживети велику експанзију.

5.1 Хардвер будућности

Интерактивна стакла би требала да представљају површине које мењају данашње таблет уређаје или мобилне телефоне, као и велике интерактивне површине. Такође уређаји за праћење покрета ће постати саставни део основних хардверских компоненти у скорој будућности. Сада смо већ сведоци паметних телевизора који прате покрете и извршавају своје, посебно написане интерактивне апликације. Такође ускоро ће бити у продаји велики број лап топ рачунара са уграђеним кинект сензорима. Дакле из угла хардвера можемо рећи да ће се развој кретати ка великим мулти функционалним површинама које су у стању да у потпуности прате и прилагоде се кориснику.



Слика 5-1 Интерактивне стаклене површине

Такође једна од битнијих ставки је интеракција више уређаја међусобно. Сада ако имате садржај на једном уређају уколико желите да га користите на више неопходно је да користите неке од cloud технологија. Међутим будућност развоја ће ићи ка томе не да је садржај заједнички него да је и могућност интеракције комбинована. На тај начин таблетом или телефоном је могуће управљати телевизором без употребе даљинског или радити са колегама заједно на пројекту без потребе да физички седите на једном месту. На овај начин се постиже већа интеракција, побољшава тимски рад и подстиче креативност.



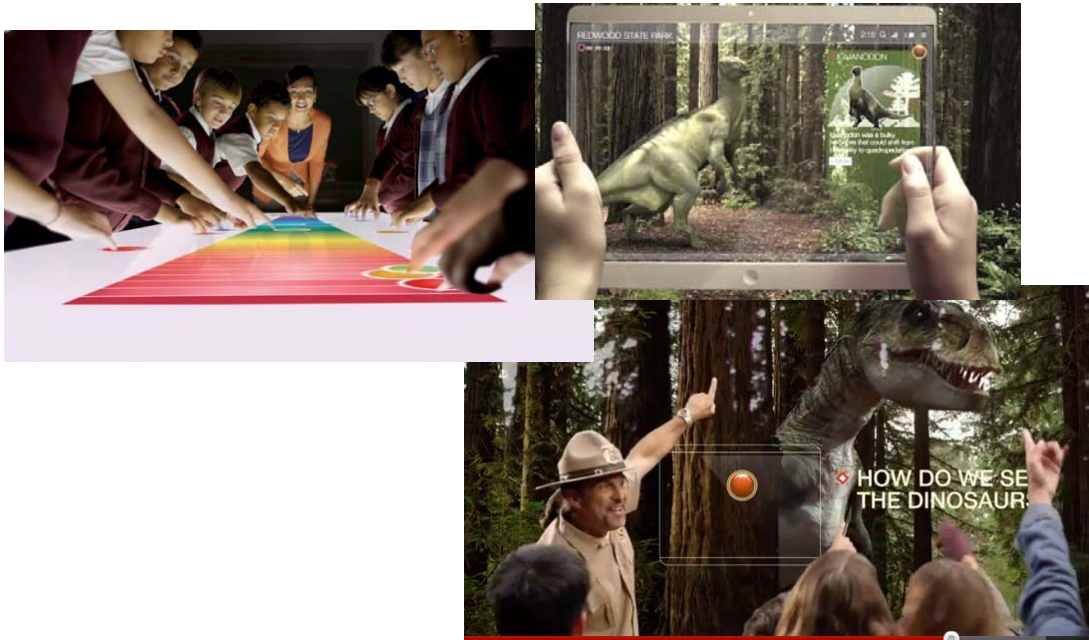
Слика 5-2 Уређаји са уграђеним сензорима покрета у међусобној интеракцији са таблет уређајима

5.2 Правац развоја софтвера

Смер развоја апликација у ближој и даљој будућности ће се у сваком случају одвијати ка бољој и дубљој интеракцији корисника са апликацијом. Апликације би требало да буду интуитивније, да брже и лакше врше интеракцију са корисником и да оно што је најбитније приказују више и боље информације. Сматра се да ће се тај развој одвијати у више различитих смерова:

- Интеракција великог броја корисника са једном површином - садашње технологије екрана осетљивих на додир могу да прате мали број корисника или до 10 прстију али циљ је да цела одељења или групе на факултету могу да раде на једној интерактивној површини.
- Надграђена стварност следеће генерације - боља интеракција са тродимензионалним објектима и заједнички садржај за већи број корисника.
- Аватари и гејмификација - употреба аватара у свету игрица је учестала и заправо постала је стандард међутим следећи корак може бити у интеракцији корисника са интерактивним изложима и излагањима на већим површинама. Зашто би се ово примењивало? Зато што надграђена стварност са самим корисником има само одређени број могућности док аватари могу пружити комплетнији утисак интерактивности, персонализација јесте мања него када корисник види свој лик директно али овај процес би се решавао из угла самог game-play-a апликације.
- Праћење покрета великог броја корисника на јавним површинама - ово се може у великој мери искористити за маркетиншке кампање, концерте, спортске догађаје или друге врсте јавних окупљања.

Неки од ових праваца су пројекти на којима се ради већ одређени број година и из компанија које раде на њима можемо видети како их они замишљају. Примери следеће генерације природних корисничких интерфејса су приказани на следећој слици.



Слика 5-3 Правци развоја софтвера у будућности

Видели смо да се ове технологије могу искористити у великом број различитих свера и области, преко образовања, маркетинга, војне индустрије, спорта и наравно индустрије игрица. Међутим заједнички закључак се намеће а то је да све ове технологије имају један заједнички циљ а то је да нам помогну и учине живот лепшим и занимљивијим.

Оно што као аутор ово рада могу рећи а то је да се заиста радујем све већој употреби ових технологија у друштву. Сви смо били свесни масовног и често штетног утицаја технологије на социјалне и здравствене аспекте друшта а сада се по први пут заиста реч природно враћа у први план. Технологије не можемо, не желимо и није потребно да је потиснемо, него је требамо искористити да наше време проводимо боље, да брже учимо и да је искористимо да нас спаја. А с обзиром да смо већ успели да подигнемо децу из фотеља и да их натерамо да се играју и скачу испред кинекта могу да тврдим да смо на правом путу!

6 ЗАКЉУЧАК

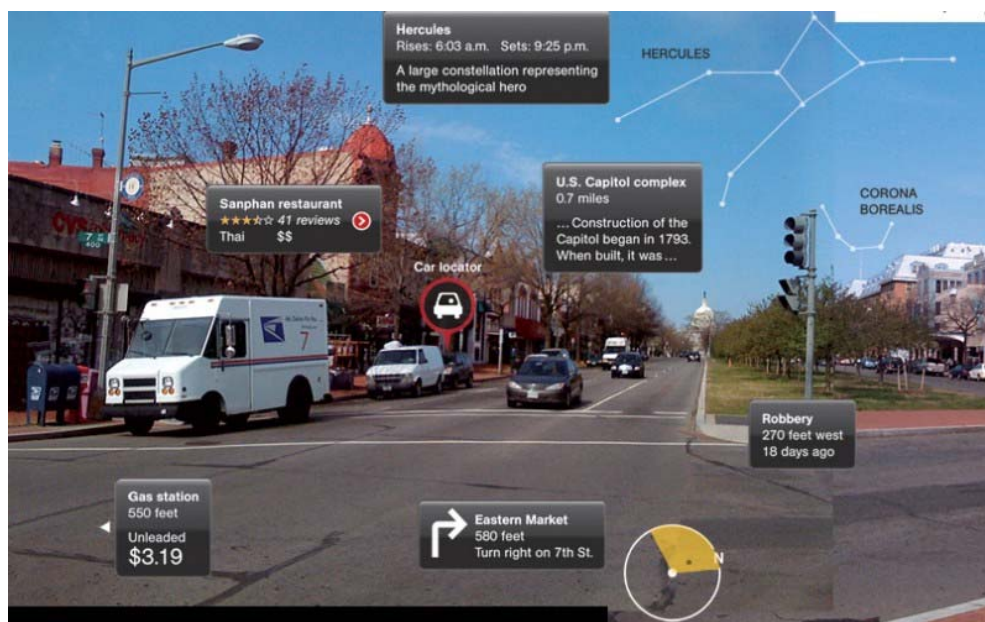
Природни кориснички интерфејси потпомогнути надграђеном стварности делују као поприлично футуристичке технологије, али смо у овом раду показали да то заправо није тако и да у оне у великој мери и добар број година присутне на сметској информатичкој сцени.

Међутим ако погледамо потенцијал ових технологија можемо рећи да њихова искоришћеност је веома мала и да из овог угла чак нисмо ни у могућности да сагледамо скалабилност употребе истих. Оно што је сигурно је да будућност примене и развоја можемо разматрати из различитих углова.

Зато можемо рећи да пословне апликације 21 века ће тек доживети своју праву трансформацију и постаће ближе корисницима, једноставније за употребу са већим бројем корисних информација. Једноставно речено постаће природније!

7 ДОДАТАК А - НАДГРАЂЕНА СТВАРНОСТ (AR - AUGMENTED REALITY)

Надграђена стварност је жив, директан или индиректан поглед на физичко, стварно животно окружење чији су елементи надграђени са компјутерски генерисаним садржајем. Ове информације могу бити звукови, видео, графика или ГПС подаци. Повезана је са много генералнијим концептом који се назива *mediated reality* у коме се поглед на стваран свет модификује рачунаром. Као резултат технологија функционише као појачање тренутне перцепције стварности.



Слика 7-1 Пример функционисања апликације која користи надграђену стварност

Разлика између виртуалне и надграђене стварности је у томе што виртуални свет у потпуности замењује стваран свет симулираним. Уз помоћ надограђене стварности информације и свету који нас окружује постају интерактивне и дигитално манипулативне.. Вештачке информације о окружењу могу се у слојевима приказивати преко погледа на стваран свет.

За примену надграђене стварности су нам потребне класичне рачунарске компоненте потпомогнуте сензорима и другим врстама улазних уређаја. Многи мобилни уређаји попут паметних телефона или таблета садрже елементе као што су камере, MEMS сензоре тип акцелометаре, ГПС, жirosкопе итд.[11]

За пројекцију је могуће користити малтене све врсте дисплеја или пројектора без обзира на величину или начин употребе. Оно што је специфично је да се ова технологија може користити и на не типичним уређајима као што су војни визири, наочаре, сочива и др.

Са софтверске стране кључна јединица мере је у томе колико реалистично су у стању да интегришу промене у стваран свет. Софтвер се мора базирати на координатама из стварног света које не зависе од камере до камере која снима слику. Овај процес се назива регистрација слике и најчешће се користи код видео праћења. Многе методе компјутерске визије надграђене стварности се базирају на визуелној одометрији. Обично ове методе се састоје из два дела:

- Прво је потребно детектовати тачке од интереса или маркере положаја и слици са камере. За овај корак се могу користити разне методе напредне детекције или процесирања слике као што су: corner detection, blob detection, edge detection или thresholding.
- Друга фаза прерачунава координате стварног света добијене из података са сензора. Неке методе захтевају објекте са познатом геометријом које се пројектују на маркере положаја. У тим случајевима често је потребно структуру сцене претходно прерачунати, а уколико није позната цела сцена или њен добар део користе се различите методе за одређивање недостајућих података.

7.1 Примена надграђене стварности у различитим областима

Надграђена стварност има велики број апликација и значајан број области може имати користи употребе исте у њој. Надграђена стварност је као и велики број других технологија првобитно био коришћен у војне сврхе, а затим у индустрији или медицинским апликацијама. Међутим убрзо је добила комплетну употребу и у комерцијалне сврхе поготово у индустрији забаве. Заправо апликације које користе надграђену стварност у великој мери су допринеле популарности iPhone.



Слика 7-2 Пример анимације свемирске летелице која је позиционирана у односу на маркер који корисник држи.

До сад ова технологија је коришћена у следећим областима:

- Археологији - Користе се за позиционирање праисторијских објеката на модерне пејзаже ради доношења закључака о даљим истраживањима.
- Архитектура - Могуће је приказивати нове не изграђене објекте на површинама које су замишљене или приказивати планове зидних инсталација у односу на посматрани зид итд.
- Уметност - омогућава особама које су хендикепиране да се баве уметношћу тако што ће им омогућити да преко техника праћења ока цртају по површинама.

- **Образовање** - Ова технологијама омогућава обогаћивање курикулума, тако што пружа ученицима садржај који је богати и боље и на реалнији начин објашњава градиво у реалном времену.
- **Медицини** - Већ сада се у великој мери користе апликације које хирурзима омогућавају приказ додатних података о стању корисника или о резултатима који су битни приликом операције. Као што су рендгенски снимци, притисак, откуцаји срца и др.
- **Војска** - Симулиране мапе и прикази положаја потенцијалних опасности или пријатељских војника су од круцијалног значаја на бојном пољу. Они се пројектују у визире корисника.
- **Спорт** - Сведоци смо великог броја преноса различитих спортских догађаја који користе ове технологије за приказивање раздаљине до гола, офсајда и многих других.
- **Навигација** - Смернице приликом кретања како у аутомобилу или приликом ходања могу пружити огроман број информација, како комерцијалних, где има добрих ресторана, где су акције и слично. До веома корисних информација као стање на путевима, временски услови или правац.
- **Туризам** - Велики број светских музеја или метропола које имају старе четврти користе нове начина дочаравања функционалности својих експоната или приказа изгледа града из различитих доба.

Надграђена стварност је већ дуго позната као појам и користе се у различитим областима већ скоро тридесет година, али са развојем природних корисничких интерфејса њихова примена ће се значајно повећати и постаће све више приметна и присутна у стварним животним ситуацијама. Цене уређаја који је користе драстично падају а са све већим употребама у маркетиншке сврхе и корисници који немају пуно новца или најнапреднију технологију је могу користити и увидети њене предности.

8 ЛИТЕРАТУРА

1. Archibald, J.A. , "Computer Science education for majors of other disciplines", (Maj 1975).
2. Steve Mann, USENIX-98, New Orleans Јун 15–19, 1998
3. Natural User Interface in .Net- Joushua Blake 2012
4. Natural User Interface (NUI): a case study of a video based interaction technique for a computer game - M. Rauterberg (1997)
5. <http://en.wikipedia.org/wiki/Kinect>
6. <http://www.softkinetic.com/en-us/solutions/depthsensecameras.aspx>
7. <http://prezi.com/>
8. Zichermann, Gabe; Cunningham, Christopher , Август 2011
9. Carol Emert, Web Advertisers Get New Tool SF Chronicle, Октобар 19, 1998
10. <http://www.retargetingnews.com/wp-content/uploads/2011/05/Retargeting-3-642x500.jpg>
11. Graham, M., Zook, M., and Boulton, A. "Augmented reality in urban places: contested content and the duplicity of code." , 2012