

Slovenská technická univerzita

Fakulta informatiky a informačných technológií

Ilkovičova 3, 842 16 Bratislava 4

Emotion Log

Tímový project

Dokumentácia k inžinierskemu dielu

Bc. Michal Biroš
Bc. Tomáš Caban
Bc. Tomáš Kunka
Bc. Filip Staňo
Bc. Tomáš Lekeň
Bc. Milan Martinkovič
Bc. Bálint Szilva

Obsah

1	Úvod	1-1
1.1	Účel dokumentu	1-1
1.2	Výber implementačného jazyka, prostredia a technológií.....	1-1
1.2.1	Serverová časť	1-1
1.2.2	Klientska časť.....	1-1
2	Prvý šprint – Audi	2-1
2.1	Používateľ je sledovaný pri práci na počítači.....	2-1
2.1.1	Analýza východzieho stavu	2-2
2.1.2	Analýza softvéru na sledovanie pohľadu	2-6
2.1.3	Analýza existujúcich riešení na rozpoznávanie emócií	2-9
2.1.4	Analýza detekovateľných emočných stavov z tváre	2-11
2.1.5	Analýza možností využitia mikrofónu pre získavanie informácií o emóciách, identite používateľa a prítomnosti používateľa	2-15
2.2	Modely sú rozpoznávané.....	2-17
2.2.1	Analýza možností strojového učenia.....	2-17
2.3	Používateľ dostane odporúčanie	2-18
2.3.1	Analýza odporúčaní pre jednotlivé emočné stavy	2-18
2.4	Zhrnutie šprintu	2-20
3	Druhý šprint – Bentley.....	3-1
3.1	Návrh	3-1
3.2	Implementácia	3-1
3.3	Zhrnutie šprintu	3-3

1 Úvod

1.1 Účel dokumentu

Tento dokument predstavuje projektovú dokumentáciu k softvérovému produktu, ktorého cieľom je pozorovať správanie používateľa pri používaní počítača, analyzovať jeho emočný stav a navrhovať odporúčania na základe analyzovaného stavu. Tento projekt vzniká na predmete Tímový projekt pod vedením Ing. Martina Labaja.

Naším cieľom je obohatiť prebraté softvérové riešenie o sledovanie používateľa prostredníctvom kamery a mikrofónu a rozšíriť tak možnosti analyzovania jeho emočného stavu.

1.2 Výber implementačného jazyka, prostredia a technológií

Implementácia je rozdelená do 2 častí:

- serverová časť
- klientska časť

1.2.1 Serverová časť

Jadro servera : Windows Server 2008

Databáza : SQL Server 2008 R2

Verziovanie a komunikácia s klientskou časťou: Team Foundation Server 2010 a IIS 7.0

1.2.2 Klientska časť

Programovací jazyk – C #

Vývojové prostredie – Microsoft Visual Studio 2010

2 Prvý šprint – Audi

Hlavnou úlohou prvého šprintu bolo oboznámiť sa s problémovou oblasťou a s prebratým softvérovým riešením. Cieľom projektu je softvérový produkt, ktorý bude schopný na základe pozorovania používateľa kamerou modelovať jeho emocionálny stav a na jeho základe mu odporúčať rôzne akcie. V rámci šprintu sme identifikovali nasledovné príbehy:

- Používateľ je sledovaný pri práci na počítači
- Používateľ je modelovaný
- Modely sú rozpoznávané
- Používateľ dostane odporúčanie

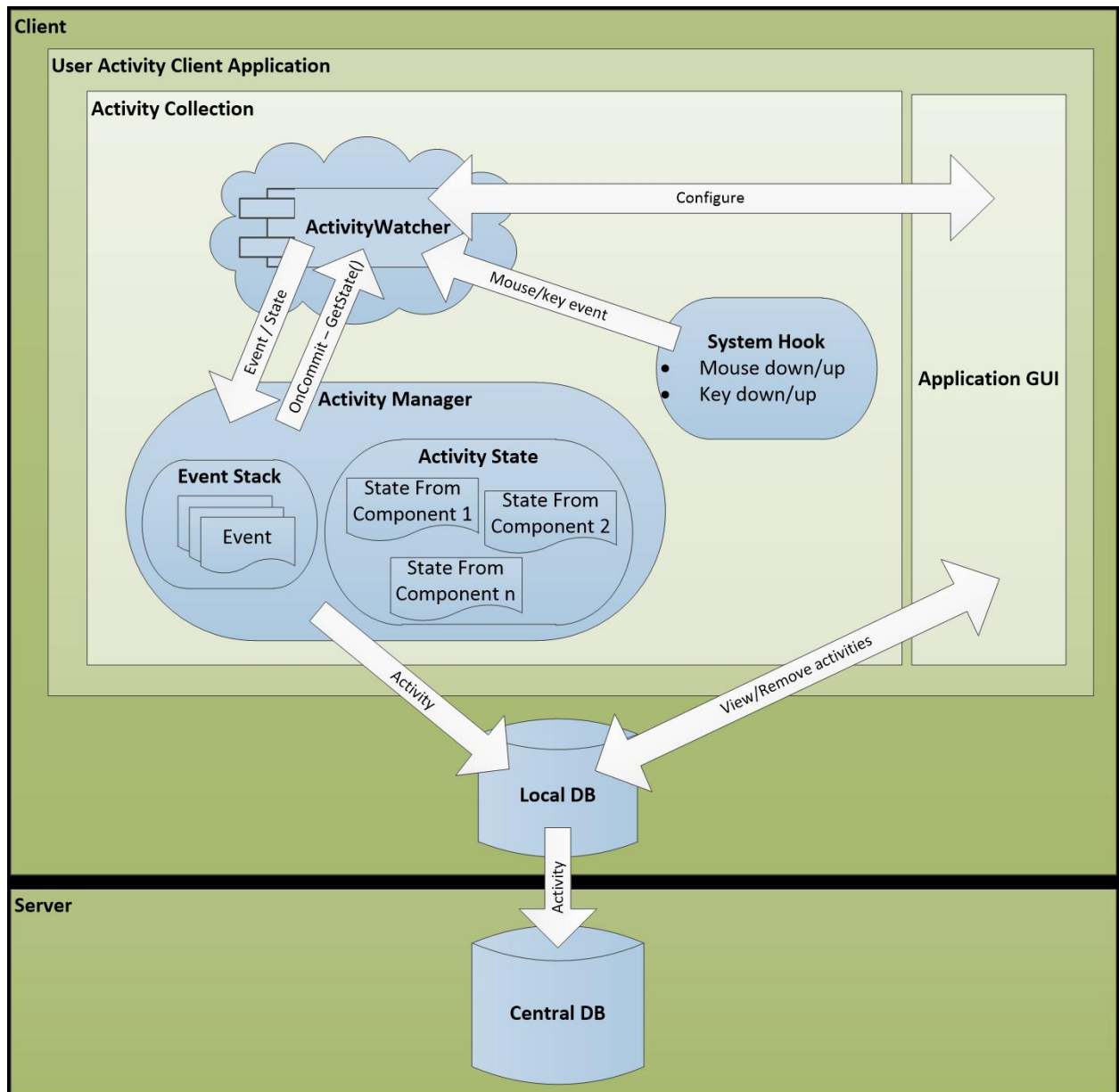
2.1 Používateľ je sledovaný pri práci na počítači

Ako používateľ som sledovaný pri práci na počítači pomocou kamery a mikrofónu.

2.1.1 Analýza východzieho stavu

2.1.1.1 Architektúra systému

Aplikáciu budeme vytvárať v už existujúcom prostredí *logovača*, ktorý zaznamenáva aktivitu používateľa na počítači. Tieto údaje následne posiela do databázy na serveri.



Obrázok 1 Architektúra existujúceho system

Chyba! Nenašiel sa žiaden zdroj odkazov. znázorňuje architektúru tohto existujúceho systému. Tvoria ho dve časti – klientská a serverová časť.

Logovač zachytáva stavy systému a aplikácií (Activity state) a udalosti (Event), ktoré nastali počas nejakej aktivity (Activity). Aktivity sa ukladajú do lokálnej databázy a v určitom časovom intervale sú odosielané na server.

2.1.1.2 Zdroje zbieraných údajov

Zdroje, z ktorých sú zbierané údaje:

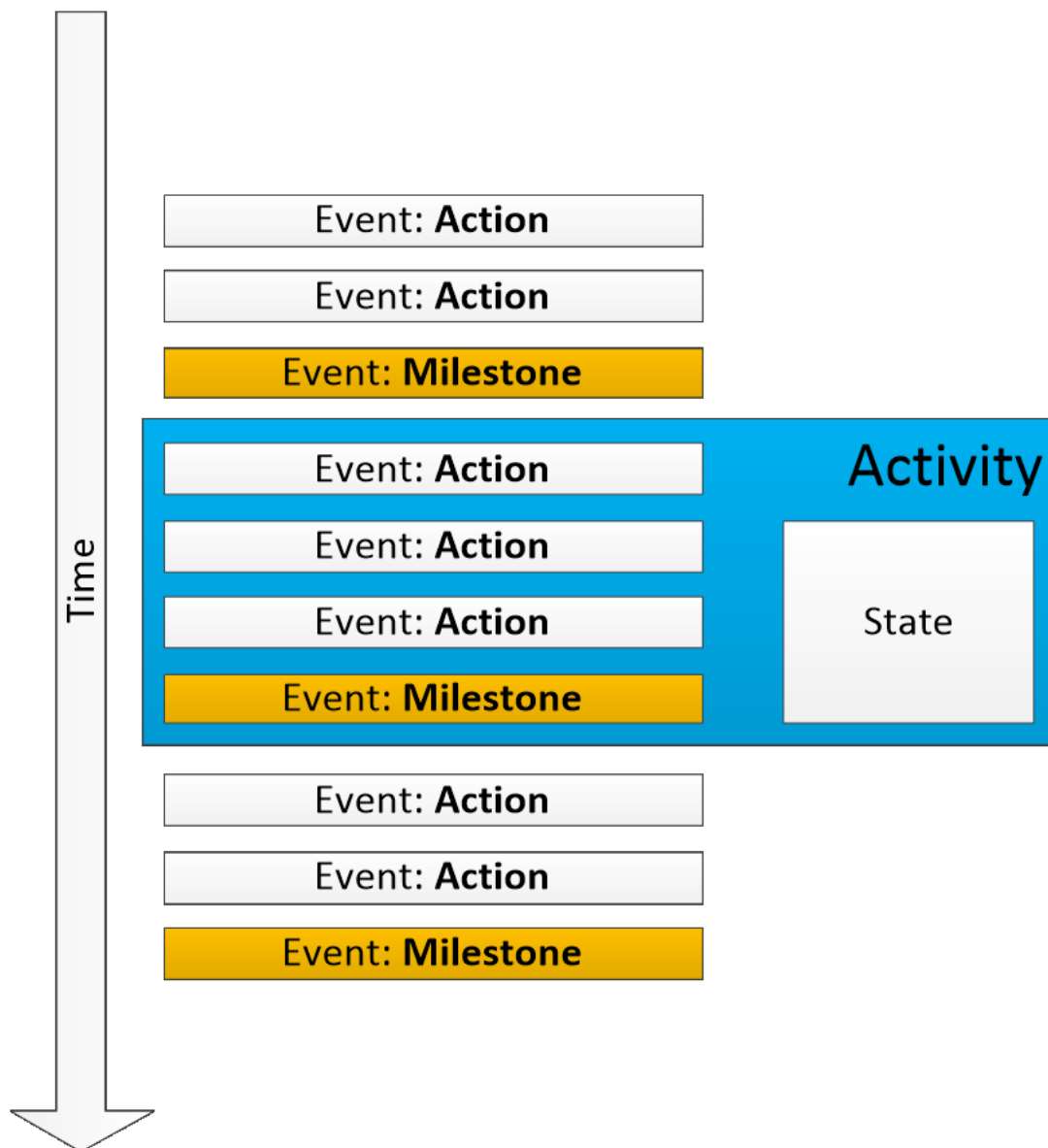
- Operačný systém – vyťaženie procesora, obsadenie operačnej pamäte
- MS Visual Studio 2012 – operácie nad projektom, operácie nad dokumentom, aktuálne otvorené okno, pozíciu v kóde
- MS Office 2010 OneNote – activity v otvorených poznámkových blokoch
- Spustené aplikácie a ich okná
- Klávesnica – stlačené klávesy
- Myš – pohyb myši po obrazovke
- Webový prehliadač (Firefox) – otvorené web stránky v jednotlivých kartách, operácie nad záložkami
- MS Lync 2010 – stav používateľa (online, offline, zaneprázdnenosť)
-

2.1.1.3 Aktivity

Údaje sú ukladané do databázy vo forme tzv. aktivít. Jedna aktivita obsahuje časovo usporiadané akcie a stavy akcií, ktoré nastali počas priebehu tejto aktivity.

```
<?xml version="1.0" encoding="utf-16"?>
<Activity
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    StartTime="2012-11-13T02:42:20.3489129Z"
    EndTime="2012-11-
13T02:42:21.3989094Z"
    Workstation="REXXAR8"
    ActivityId="96ff3b03-004d-4ea4-983e-4acce11296ee"
    xmlns="http://url/ActivityService">
  <Events>
    <Event>
      <EventState />
    </Event>
    <Event />
    <Event />
    ...
  </Events>
  <States>
    <State />
    <State />
    ...
  </States>
</Activity>
```

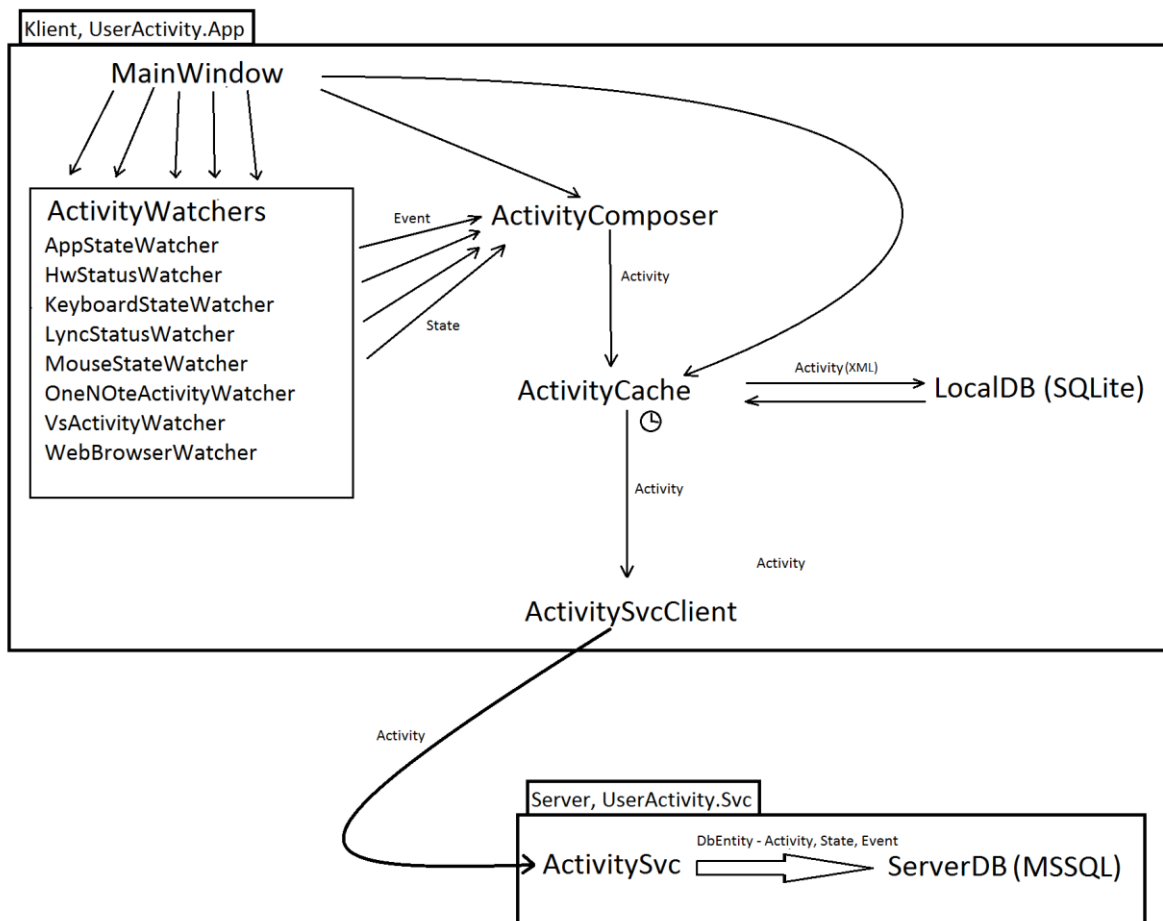
Zdrojový kód 1 Príklad aktivity zapísanej v XML



Obrázok 2 Aktivita obsahuje časovo usporiadané akcie. Aktivita je ukončená práve jednou akciou, ktorá je míľnikom.

Chyba! Nenašiel sa žiaden zdroj odkazov. znázorňuje, ako sa vytvárajú aktivity v čase. ystém zaznamenáva akcie z jednotlivých *watcherov*. Ak je zaznamenaná akcia s atribútom *milestone*, aktivita je uzavretá a uložená do lokálnej databázy. Následne sa vytvorí nová prázdna aktivita.

2.1.1.4 Priebeh zaznamenávania údajov



Obrázok 3 Schéma priebehu zaznamenávania aktivít

Chyba! Nenašiel sa žiaden zdroj odkazov. znázorňuje priebeh zaznamenávania aktivít od pustenja programu až po uloženie do databázy na serveri.

MainWindow – počiatočný komponent. Na začiatku inicializuje inštancie jednotlivých *ActivityWatchers*, *ActivityComposer*, *ActivityCache*.

ActivityWatchers – priebežne zaznamenávajú jednotlivé aspekty práce na počítači. Implementujú interfejs *IActivityWatcher*. Ak nejaký *ActivityWatcher* zachytí udalosť (*Event*), odošle ju do triedy *ActivityComposer*.

ActivityComposer – zaznamenáva jednotlivé *Event* z *ActivityWatcher*. Ak zaznamená *Event* s atribútom `isMilestone = true`, zavolá metódu `OnActivityFinishing()` v každom *ActivityWatcher*. Táto metóda vracia stav sledovania (*State*).

ActivityComposer vytvorí z týchto údajov *Activity*, ktorú pošle triede *ActivityCache*.

ActivityCache – zaznamenáva *Activity* a ukladá do lokálnej databázy vo forme XML. Prevod triedy *Activity* do XML je vykonávaný pomocou triedy *XmlSerializer*.

ActivityCache v pravidelnom intervale vyberá aktuálne aktivity z lokálnej databázy a preposiela ich na server pomocou triedy *ActivitySvcClient*.

ActivitySvcClient – zabezpečuje spojenie klientskej aplikácie so službou na serveri (*ActivitySvc*). Pomocou metódy *CommitActivity*

ActivitySvc – služba, ktorá beží na serveri a prijíma *Activity*, odosielané komponentom z klientskej aplikácie.

Z jednotlivých aktivít typu *Activity* vyberie udalosti (*Event*) a stavy (*State*), ktoré uloží do databázy na serveri. Pred uložením sú tieto entity prekonvertované na databázové entity pomocou konvertorov (*Convertors*).

2.1.2 Analýza softvéru na sledovanie pohľadu

Sledovanie pohľadu je pre nás dôležité najmä z hľadiska možnosti určenia miery sústredenia pozorovaného používateľa. Dôležitými atribútmi sú jednak primeraná presnosť, licencia bez poplatkov pre akademické účely ale aj možnosť exportu údajov o sledovaní a nízka náročnosť na použitú kameru.

2.1.2.1 ITU Gaze tracker

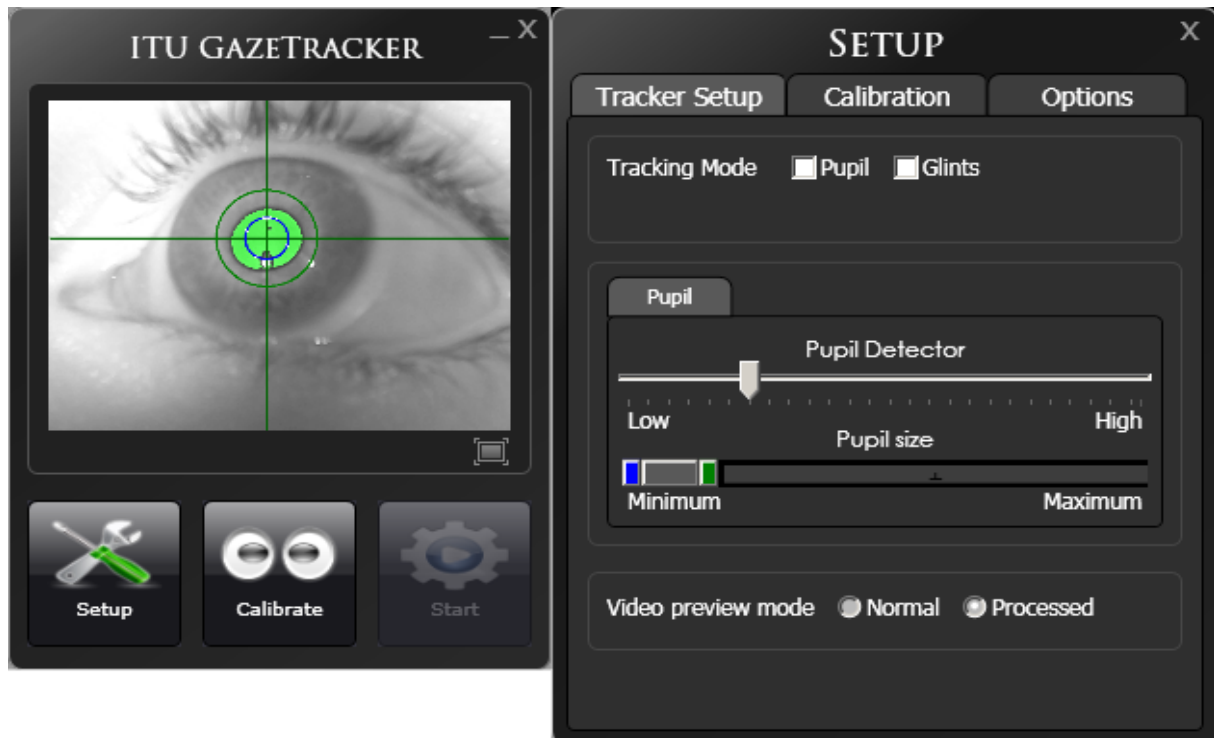
Jedná sa o nekomerčné riešenie. Spolupracuje s kamerami, ktoré obsahujú infračervené diódy ale aj s takými, ktoré ich nemajú.

Výhody:

- vhodná presnosť sledovania pohľadu
- možnosť exportovania údajov o sledovaní pohľadu

Nevýhody:

- nutná kalibrácia a to aj po miernej zmene polohy používateľa
- stand-alone riešenie bez otvoreného kódu
- potrebná konfigurácia kamery cez používateľské rozhranie
- pre maximálnu presnosť je potrebné použitie s kamerou, ktorá podporuje vysielanie infračerveného svetla



Obrázok 4 Ukážka používateľského rozhrania ITU Gaze tracker

2.1.2.2 Camera Mouse 2012

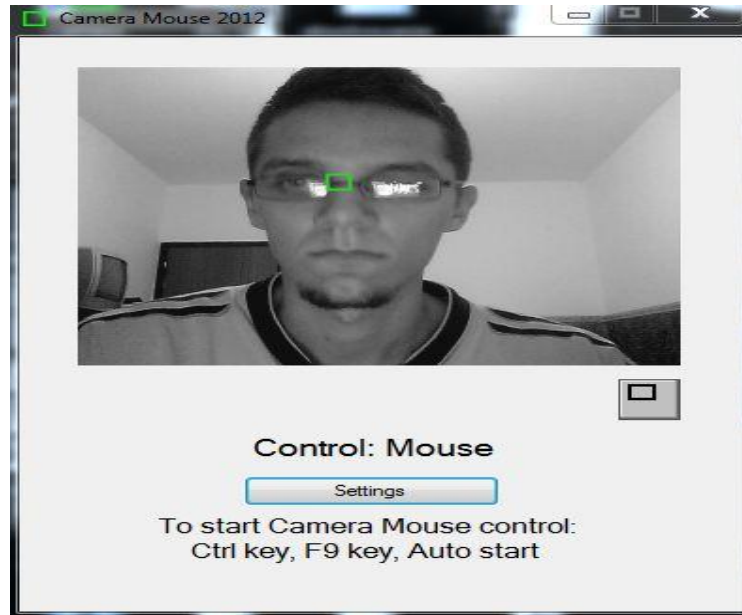
Taktiež nekomerčné riešenie, ktoré sa zameriava na ovládanie kurzora myši pomocou pohľadu.

Výhody:

- nie je potrebná zdĺhavá kalibrácia
- samo detekcia pevného bodu na tvári, pomocou ktorého je zachytávané natočenie hlavy (zelený bod na obrázku predstavuje pevný bod, ktorý bol nájdený)
- vysoká presnosť

Nevýhody:

- je síce zadarmo ale zdrojový kód nie je k dispozícii
- žiadny export údajov



Obrázok 5 Ukážka Camera Mouse 2012

2.1.2.3 Track Eye

Výhody:

- open source
- bez kalibrácie

Nevýhody:

- nepresné
- pre zlepšenie presnosti je potrebná zložitá konfigurácia nastavení, po ktorej je presnosť na stále nízkej úrovni

2.1.2.4 Zhodnotenie analýzy

Analyzované boli aj ďalšie riešenia, tie však neuvádzame kvôli ich podstatnej miere negatívnych vlastností ako napríklad nestabilita, nepresnosť ale aj nízka kompatibilita s najpoužívanejšími platformami. Na základe analyzovaných riešení sme sa rozhodli funkcionality sledovania pohľadu nezahrnúť do nášho projektu. Hlavnými dôvodmi boli uzavretosť kódu daného softvéru, poprípade nízka úroveň riešení s otvoreným kódom. Ďalším dôležitým negatívnym faktorom bola pri väčšine riešení nutnosť modifikácie rôznych nastavení ale aj kalibrovanie sledovania. Tieto vlastnosti prispievali k nízkej úrovni používateľsky príjemného prostredia, ktoré by mohlo mať za následok frustráciu používateľa, čo je pravý opak toho o čo sa v našom projekte snažíme.

Na základe analýzy sme však napríklad identifikovali náležitosti, ktoré sú predpokladom pre zlepšenie presnosti a kvality sledovania používateľa. Jedná sa napríklad o vhodné

umiestnenie kamery. Lepšie výsledky boli dosahované pri externých kamerách, ktoré boli umiestnené na stole pod úrovňou tváre používateľa na rozdiel od vstavaných kamier v obrazovke. Ďalšími aspektmi sú vhodné osvetlenie či kvalita záznamu, ktorý kamera produkuje.

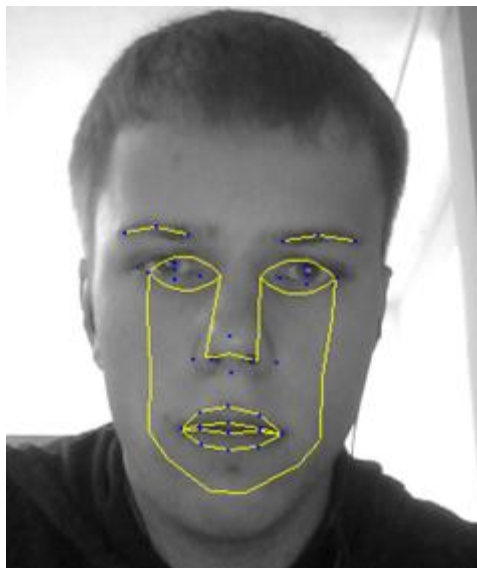
2.1.3 Analýza existujúcich riešení na rozpoznávanie emócií

2.1.3.1 FaceAPI

Rozoznáva základné črty tváre priamo z webkamery a v pomerne nízkom časovom intervale identifikuje jednotlivé časti tváre a vyznačí ich.

Vyznačuje:

- 3 body obočia (definuje tvar)
- 4 body nosa
- 5 bodov očí
- 14 bodov úst

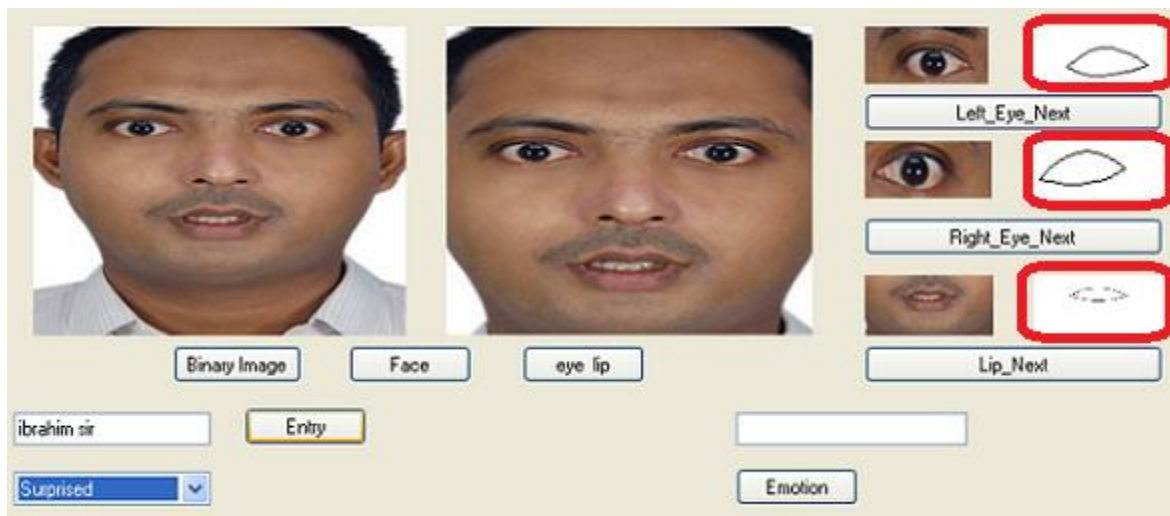


Obrázok 6 Spracovanie tváre pomocou FaceAPI

Nekomerčná licencia neobsahuje kompletne služby, vývojárska licencia obsahuje kvalitné výstupy aj plnohodnotné spracovanie tváre – treba o ňu požiadať. Optimálne je získať výstup súradníc pre jednotlivé časti tváre.

2.1.3.2 Human Emotion Detection

- Projekt bangládešského študenta
- CPOL licencia
- Rozoznáva emóciu podľa tvaru očí a úst
- Pomocou prevodu obrazu tváre do binárneho prostredia segmentuje jednotlivé časti tváre(oči a pery), z ktorých následne odvádza emóciu.
- Naprogramovaný v C# pomocou .NET.
- Rok vydania 2010.



Obrázok 7 GUI aplikácie Human Emotion Detection

Červenou farbou je zvýraznený využiteľný výstup tvaru očí a úst.

2.1.3.3 Face.com

Rozpoznanie viacerých tvárí z fotografie. Identifikácia jednotlivých bodov a charakteristík nálady, či veku.

Vyznačuje:

- 1 bod nosa
- 2 body očí
- 3 body úst

Attributes:	
age_est:	45 (47%)
age_min:	39 (47%)
age_max:	53 (47%)
face:	true (99%)
gender:	male (91%)
glasses:	false (95%)
lips:	sealed (65%)
mood:	happy (63%)
smiling:	true (74%)
Rotations:	
roll:	-0.64°
yaw:	-1.4°
pitch:	-6.08°

Obrázok 8 Výstupy aplikácie Face.com

Pokúša sa o určenie:

- Veku
- Pohlavia
- Otvorenia pier
- Nálady
- Úsmevu
- Okuliarov

Ku všetkým spomínaným hodnotám sa pokúša prideliť pravdepodobnosť.

Verejné API na serveri bolo uzavreté v októbri 2012, avšak použiteľné knižnice sú na stiahnutie vo viacerých jazykoch (C#, JAVA, PHP, Python, RUBY, JS ...)

2.1.4 Analýza detekovateľných emočných stavov z tváre

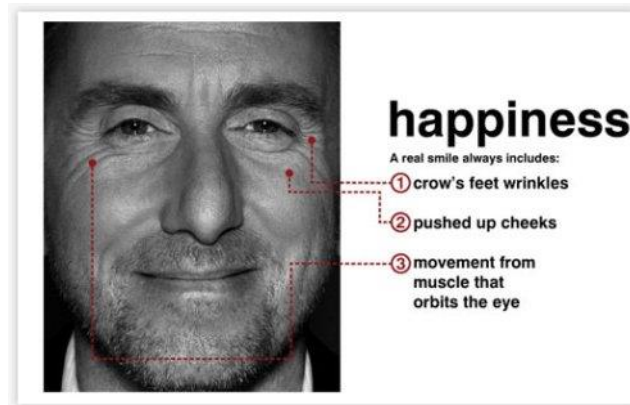
Existujúce rozpoznateľné emočné stavy:

- Radosť
- Smútok
- Hnev
- Strach
- Prekvapenie
- Znechutenie
- Pohrdanie

2.1.4.1 Radosť

Špecifické znaky:

1. Roztiahnuté a vydvihnuté kútiky úst
2. Vydvihnuté líca
3. Zvraštené vonkajšie očné svaly

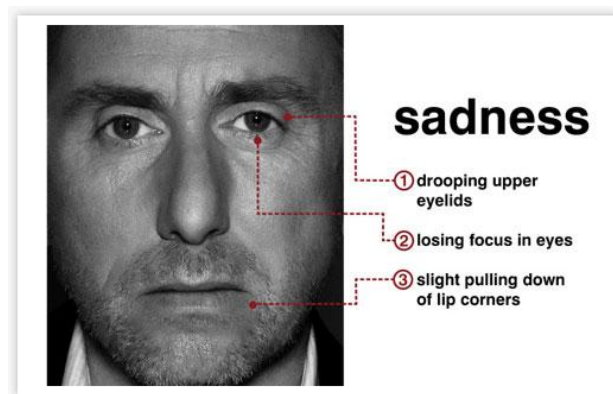


Obrázok 9 Radosť

2.1.4.2 Smútok

Špecifické znaky:

1. Mierne padnuté kútiky úst
2. Padnuté horné viečko

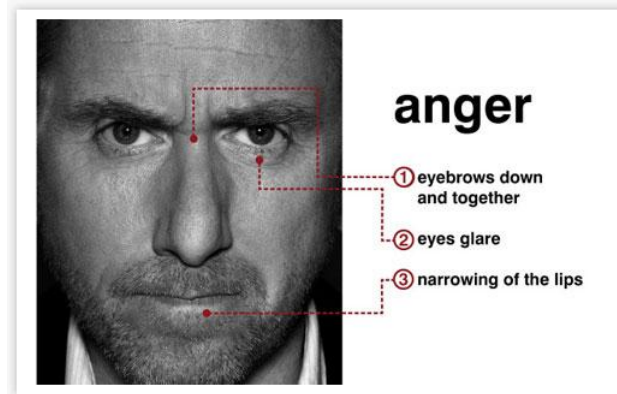


Obrázok 10 Smútok

2.1.4.3 Hnev

Špecifické znaky:

1. Zvraštené obočie – bližšie k očiam a k sebe navzájom
2. Maximálne otvorené spodné viečko
3. Vyrovnané a stiahnuté pery



Obrázok 11 Hnev

2.1.4.4 Strach

Špecifické znaky:

1. Zdvihnuté obočie, bližšie k sebe navzájom
2. Maximálne otvorené horné viečko
3. Napnuté dolné viečko
4. Pootvorené ústa, natiiahnuté pery

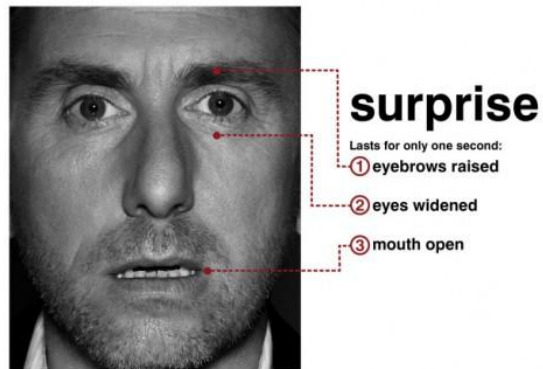


Obrázok 12 Strach

2.1.4.5 Prekvapenie

Špecifické znaky:

1. Zdvihnuté obočie
2. Pootvorené ústa
3. Maximálne otvorené oči

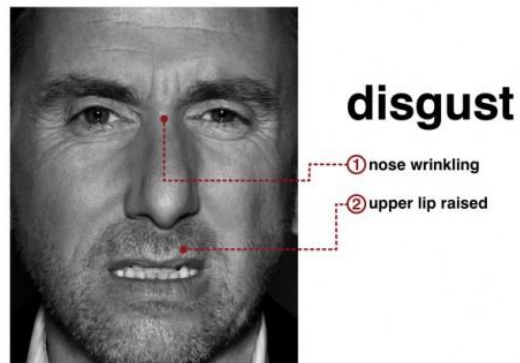


Obrázok 13 Prekvapenie

2.1.4.6 Znechutenie

Špecifické znaky:

1. Vydvihnutie hornej pery
2. Vrásky medzi očami

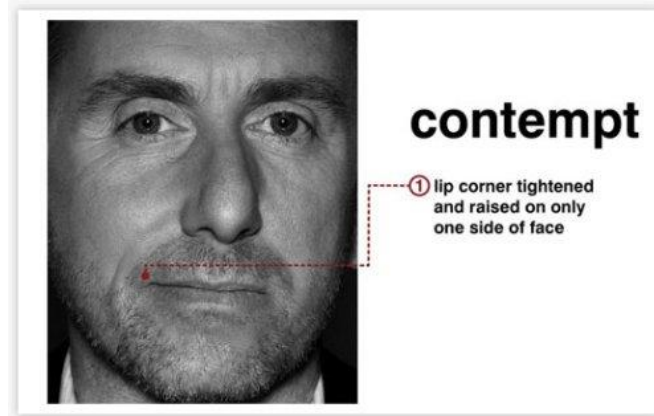


Obrázok 14 Znechutenie

2.1.4.7 Pohrdanie

Špecifické znaky:

1. Vydvihnutá pera na jednej strane – polovičný úsmev



Obrázok 15 Pohrdanie

2.1.5 Analýza možností využitia mikrofónu pre získavanie informácií o emóciách, identite používateľa a prítomnosti používateľa

2.1.5.1 Emócie

Získavanie emócií z hlasu používateľa je najmä predmetom výskumu. Existuje množstvo vedeckých článkov, ktoré sa zaoberajú touto problematikou. Nepodarilo sa nájsť žiadny komerčný nástroj, ktorý by umožňoval získavanie emócií z hlasu. Analyzovali sme dva akademické nástroje, ktoré sa snažia o získavanie emócií z hlasu.

2.1.5.2 OpenEAR

<http://sourceforge.net/projects/openart/>

Nástroj napísaný v jazyku C++. Umožňuje získavanie emócií priamo z mikrofónom zachyteného zvuku, alebo zo zvukových súborov. Je ho možné použiť ako samostatný nástroj (príkazový riadok) alebo ako knižnicu. Používa 6 modelov na vyhodnocovanie emócií, z ktorých väčšina pracuje na princípe, že majú niekoľko stavov (strach, hnev, šťastie, smútok, nechť, ...) a vyjadrujú percentuálnu pravdepodobnosť pre jednotlivé stavy.

Na vyhodnocovanie emócií sa používajú vytvorené databázy, ale je možné vytvoriť aj vlastné a tak trénovať nástroj.

Jedná sa o open-source nástroj. Nevýhodou je veľmi slabá a nekonzistentná databáza. Na projekte sa nepracuje od roku 2009.

2.1.5.3 EmoVoice - Real-time emotion recognition from speech

<http://www.informatik.uni-augsburg.de/de/lehrstuehle/hcm/projects/tools/emovoice/>

Rovnako akademický nástroj napísaný v jazyku C++. Umožňuje získavanie emócií z hlasu používateľa. Nástroj posudzuje emócie pozitívna a negatívna a vracia percentuálnu pravdepodobnosť pre danú emóciu. Je ho možné použiť ako samostatný nástroj. Poskytuje aj grafické používateľské rozhranie, ktoré umožňuje aj tréning programu pre používateľa.

EmoVoice bol integrovaný do frameworku **Social Signal Interpretation (SSI)**, ktorý poskytuje aj ďalšie funkcie, napr. rozoznávanie emócií z obrazu. SSI sa dá použiť aj ako C++ knižnica. SSI tiež poskytuje špeciálny XML editor s množstvom funkcií pre prácu s nástrojom.

Je to voľne dostupný nástroj. Jeho veľkou nevýhodou je rovnako ako pri OpenEAR veľmi slabá dokumentácia, je dostupný len jednoduchý tutorial.

2.1.5.4 Identita používateľa

Pri hľadaní možností využitia hlasu získavaného pomocou mikrofónu na identifikáciu používateľa je potrebné rozoznávať dve základné kategórie:

- Speech recognition – rozpoznanie reči
- Speaker recognition – rozpoznanie hovoriaceho

Na rozoznávanie reči existuje množstvo nástrojov a knižníc, za všetky môžem spomenúť PRAAT (<http://www.fon.hum.uva.nl/praat/>), naozaj rozsiahly projekt na analýzu reči. Niektoré algoritmy sú využívané aj už spomínaným EmoVoice nástrojom. Tieto nástroje či knižnice sú však pre náš projekt a pre naše potreby len veľmi málo využiteľné.

Táto oblasť je najmä cieľom výskumu a je veľmi komplexná. Existujú teoretické modely na rozoznávanie hovoriaceho či na základe slov, viet, ale aj hlasu. Nepodarilo sa však nájsť žiadny voľne dostupný nástroj, ktorý by sa o to aspoň snažil. Existuje niekoľko vedeckých článkov, kde aj implementujú takéto systémy, žiadny však nie je dostupný.

2.1.5.5 Prítomnosť používateľa

V našom projekte sa uvažovalo aj o získavaní informácie o prítomnosti používateľa pri počítači. Úlohou bolo analyzovať dostupné nástroje, ktoré by toto umožňovali.

2.1.5.6 Sonar Power Manager

Program, ktorý umožňuje zistiť prítomnosť používateľa na základe vysielania ultrazvuku (20 kHz). Program vypína obrazovku v prípade neprítomnosti. Program je voľne dostupný aj so zdrojovým kódom. Tento prístup nefunguje pre všetky počítače, niektoré nedokážu vysieľať ultrazvuk. Na 3 testovaných počítačoch toto riešenie nefungovalo.

2.2 Modely sú rozpoznávané

2.2.1 Analýza možností strojového učenia

Strojové učenie môžeme deliť podľa viacerých kritérií. Jedno z možných delení je nasledovné:

- Induktívne – z konkrétnych príkladov generuje všeobecnejšiu znalosť
- Deduktívne – zo všeobecnej znalosti dedukuje menej všeobecnú, lepšie prispôsobenú na riešenie konkrétneho problému

Ďalšie možné delenie je podľa spôsobu získavania trénovacích príkladov:

- Online – získavajú trénovanie príklady postupne. Toto zodpovedá inkrementálnemu učeniu
- Offline – disponujú všetkými trénovacími príkladmi súčasne. Zodpovedá neinkrementálnemu učeniu

V našom prípade sa jedná o offline učenie, keď potrebujeme najprv určitú štartovaciu množinu príkladov, na ktorých budeme môcť trénovať náš učiaci algoritmus. Postupne však budeme môcť použiť znaky online učenia, pretože by sme mali byť schopný učiť algoritmus aj z nových, postupne prichádzajúcich trénovacích príkladov.

Podľa stupňa kontroly rozlišujeme učenie na:

- Kontrolované – disponuje spätnou väzbou o úspešnosti učenia
- Nekontrolované – neexistuje spätná väzba

V našom prípade by bolo vhodné, ak by sme dostávali spätnú väzbu od používateľa, pretože by to pomohlo vylepšiť algoritmus rozpoznávania emočných stavov. Musíme však dbať na to, aby sme nezaťažovali používateľa zbytočným množstvom požiadaviek na spätnú väzbu.

Náš problém zodpovedá učeniu s učiteľom, teda priradovaniu tried k určitým objektom. Na začiatku procesu potrebujeme trénovaciu sadu príkladov, na základe ktorých sa vytvorí klasifikátor, ktorý bude následne priradovať vstupné dáta určeným triedam, v našom prípade emočným stavom.

Algoritmus učenia v našom prípade bude vhodné navrhnuť ako rozhodovací strom, ktorého listy budú predstavovať jednotlivé emočné stavy. Tento strom sa vytvorí na základe analýzy vzorov so známym stavom. Pri rozhodovaní sa bude postupovať od koreňa stromu, pričom v každom uzle sa testuje hodnota určitého znaku, až kým sa testovací príklad nedostane do niektorého z listov.

2.3 Používateľ dostane odporúčanie

2.3.1 Analýza odporúčaní pre jednotlivé emočné stavy

Cieľom analýzy je nájsť vhodné odporúčania aktivít, ktoré by mohli zmeniť náladu používateľa systém. Samozrejme ide o zmenu z negatívneho stavu do pozitívneho. Odporúčené aktivity by mali byť vhodné a použiteľné v kancelárskom pracovnom prostredí. Identifikovali sme tieto základné negatívne emócie, s ktorými sa môžeme stretnúť v práci:

- frustrácia
- obavy/nervozita
- hnev
- niečo sa nám nepáči
- sklamanie/smútok

Môžeme rozoznávať aj pozitívne stavy, ale tie u používateľa meniť nebudeme.

2.3.1.1 Spôsoby zmeny nálady

2.3.1.1.1 Fyzická relaxácia

Základnou myšlienkou je zabezpečenie uvoľnenie napätia a stresu. Medzi svalmi a náladou bolo dokázané spojenie. Dokonca keď človek používa svalstvo na ústach – usmeje sa jeho nálada sa zmení. Toto je zaujímavé zistenie, keďže prirodzene chápeme opak – ak je človek šťastný usmeje sa.

- Dýchanie:
 - frekvencia dýchania sa drasticky mení pri rôznych emočných stavoch(keď sme nahnevaní dýchame rýchlejšie, pri strachu zadržujeme dych)
 - ak sa dýchanie naučíme ovládať, dokážem ovládnuť svoje emócie
- Svalová relaxácia:
 1. posadiť sa pohodlne
 2. zavrieť oči
 3. začať relaxovať svaly – od nôh smerom nahor
 4. sústredte svoju pozornosť na dýchanie
 5. zhlboka dýchať a počítať

2.3.1.1.2 Zmena zamerania

Ak človek vykonáva určitú činnosť v neustále dookola, znižuje sa jeho pozornosť a jeho nálada sa mení na negatívnu, stáva frustrovaný. Ak sa po tejto činnosti nedostaví žiadaný výsledok, nastáva u ľudí sklamanie a smútok. Preto je potrebné občas zmeniť zameranie, ktorému sa ľudia v práci venujú. Návrhy odporúčaní pre zmenu zamerania sú:

- Hra:
 - pri hraní hier sa zameranie veľmi rýchlo mení, myseľ človeka sa sústreďuje na dosiahnutie dobrého výsledku v hre
 - hra by mala byť dostatočne náročná a poskytovať dobrý pocit z víťazstva – mozog človeka odmení dobrým pocitom ak splní nejakú úlohu
- Socializovanie:
 - ľudia prirodzene žijú v skupinách, preto ak počas dňa človek pracuje sám je dobré ak svoje problémy preberie s niekým iným
 - socializovanie sa dá spojiť aj s hrou – je vhodné mať v kancelárskom prostredí umiestnenú nejakú hru(šípky, malý basketbalový kôš)

2.3.1.1.3 Zvýšenie hladiny hormónov

Hormóny sú zlúčeniny, ktoré v tele slúžia ako chemický poslovia medzi bunkami. Sú produkované v ľudskom tele a riadia priebeh a vzájomnú koordináciu reakcií v organizme. Medzi dôležité hormóny, ktorými je možné ovplyvniť náladu sú:

- Endorfin:
 - hormón šťastia – vyplavuje sa v mozgu a spôsobuje dobrú náladu, pocit šťastia a tlmí bolesť
 - vyplavuje pri svalovej námahe, športovej činnosti
 - **Ako zvýšiť hladinu:** prechádzka, páľivé jedlá, čokoláda
- Serotonin:
 - hormón, ktorého nedostatok spôsobuje depresie a agresivitu
 - jeho hladinu je možné ovplyvniť jedlom alebo drogami
 - **Ako zvýšiť hladinu:** papája, banány, datle

2.3.1.1.4 Hudba

Hudba dokáže podľa výskumov taktiež zmeniť emocionálny stav. Vytvorí buď úplne nový emočný stav, alebo v poslucháčovi vyvolá emóciu zo spomienok(toto je typické pre hudbu z pohrebov, svadieb). Odporúčanie skladieb by mohlo byť problematické, keďže hudobný vkus je subjektívny. Môžeme sa však pokúsiť vytvoriť akési univerzálne hudobné skupiny. Určite by bolo vhodné prehrávanie motivačnej hudby, ak je používateľ frustrovaný, alebo upokojujúcej ak je nahnevaný.

2.3.1.1.5 Farba pozadia

Niekoľko starobylých kultúr, vrátane Egypta či Číny praktikovalo liečenie farbami. Dnes sa tiež používa v alternatívnej medicíne. Niektorí psychológovia sú voči terapii farbami skeptickí a považujú ju za preceňovanú. V našom prípade by sme sa mohli pokúsiť zmeniť náladu používateľa zmenou farby pozadia na počítači. Bolo však dokázané, že tento efekt je len dočasný. Vnímanie farieb je subjektívne, ale existujú farby, ktoré sú univerzálne:

- Červená, oranžová a žltá sú vnímané ako teplé farby a evokujú emócie tepla, komfortu až hnevu.
- Modrá, purpurová a zelená sú vnímané naopak ako chladné a evokujú pocity pokoja, smútku a ľahostajnosti.

2.4 Zhrnutie šprintu

Cieľom šprintu bolo oboznámiť sa s prebratým softvérovým riešením a analyzovať možné prístupy sledovania používateľa a zisťovania jeho emočného stavu pomocou kamery a mikrofónu. Taktiež sme analyzovali odporúčania pre emocionálne stavy. Z analyzovaných prístupov sme vybrali tie, ktoré sú pre potreby nášho projektu vhodné. V oblasti analýzy pomocou kamery je to projekt Human Emotion Detection a pre spracovanie pomocou mikrofónu sme vybrali projekt openEAR.

3 Druhý šprint – Bentley

Hlavným cieľom druhého šprintu bolo aplikovať analyzované riešenia, konkrétne získavanie a spracovanie obrazu z kamery. Pracovali sme aj na možnosti využiť získavanie a spracovanie zvuku v projekte a taktiež na úlohách súvisiacich s podporným softvérom. Dôležitou úlohou tohto šprintu bolo aj zostaviť dokumentáciu k riadeniu projektu a k inžinierskemu dielu.

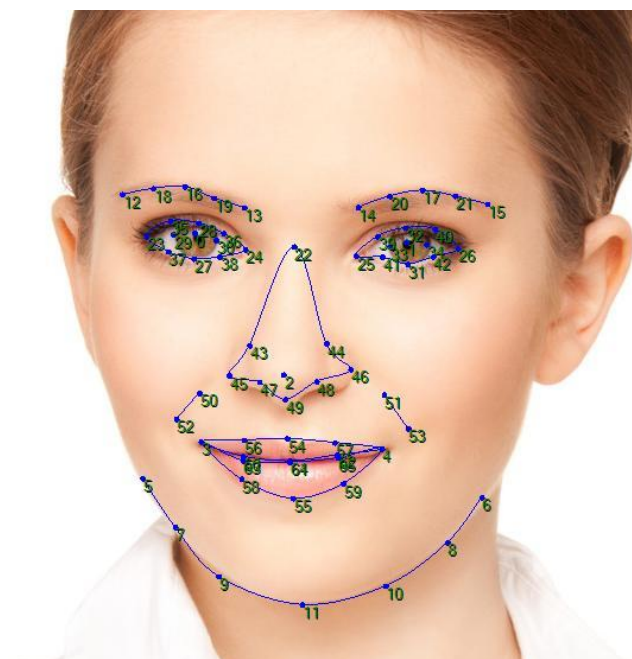
3.1 Návrh

V rámci šprintu sme mali naplánované aplikovať riešenia, ktoré sme analyzovali v predchádzajúcom šprinte. V oblasti spracovania obrazu z kamery sme však projekt HED nahradili projektom Luxand, ktorý viac vyhovuje našim potrebám. Tento projekt rozoznáva 66 bodov tváre, pričom zahŕňa oči, obočie, nos, ústa aj bradu. Poskytuje tak vhodnejšie dáta na ďalšie spracovanie.

V oblasti spracovania zvuku sme sa rozhodli nahradiť projekt openEAR, keďže tento projekt už nie je vyvíjaný a jeho dokumentácia je nekonzistentná. Namiesto neho sme sa rozhodli použiť knižnicu openSMILE, ktorú používa aj projekt openEAR.

3.2 Implementácia

Počas šprintu sme implementovali do prebratého softvérového riešenia spracovanie obrazu z kamery pomocou projektu Luxand. Luxand FaceSDK rozoznáva 66 bodov tváre, ktoré reprezentuje dvoma súradnicami.



Obrázok 16 Body tváre rozoznávané projektom Luxand

V nasledujúcej tabuľke sú uvedené všetky body, ktoré Luxand rozoznáva. K týmto bodom je možné pristupovať prostredníctvom ich názvov, uvedených v tabuľke.

Názov bodu tváre	Hodnota
FSDKP_LEFT_EYE	0
FSDKP_RIGHT_EYE	1
FSDKP_LEFT_EYE_INNER_CORNER	24
FSDKP_LEFT_EYE_OUTER_CORNER	23
FSDKP_LEFT_EYE_LOWER_LINE1	38
FSDKP_LEFT_EYE_LOWER_LINE2	27
FSDKP_LEFT_EYE_LOWER_LINE3	37
FSDKP_LEFT_EYE_UPPER_LINE1	35
FSDKP_LEFT_EYE_UPPER_LINE2	28
FSDKP_LEFT_EYE_UPPER_LINE3	36
FSDKP_LEFT_EYE_LEFT_IRIS_CORNER	29
FSDKP_LEFT_EYE_RIGHT_IRIS_CORNER	30
FSDKP_RIGHT_EYE_INNER_CORNER	25
FSDKP_RIGHT_EYE_OUTER_CORNER	26
FSDKP_RIGHT_EYE_LOWER_LINE1	41
FSDKP_RIGHT_EYE_LOWER_LINE2	31
FSDKP_RIGHT_EYE_LOWER_LINE3	42
FSDKP_RIGHT_EYE_UPPER_LINE1	40
FSDKP_RIGHT_EYE_UPPER_LINE2	32
FSDKP_RIGHT_EYE_UPPER_LINE3	39
FSDKP_RIGHT_EYE_LEFT_IRIS_CORNER	33
FSDKP_RIGHT_EYE_RIGHT_IRIS_CORNER	34
FSDKP_LEFT_EYEBROW_INNER_CORNER	13
FSDKP_LEFT_EYEBROW_MIDDLE	16
FSDKP_LEFT_EYEBROW_MIDDLE_LEFT	18
FSDKP_LEFT_EYEBROW_MIDDLE_RIGHT	19
FSDKP_LEFT_EYEBROW_OUTER_CORNER	12
FSDKP_RIGHT_EYEBROW_INNER_CORNER	14
FSDKP_RIGHT_EYEBROW_MIDDLE	17
FSDKP_RIGHT_EYEBROW_MIDDLE_LEFT	20
FSDKP_RIGHT_EYEBROW_MIDDLE_RIGHT	21
FSDKP_RIGHT_EYEBROW_OUTER_CORNER	15
FSDKP_NOSE_TIP	2
FSDKP_NOSE_BOTTOM	49
FSDKP_NOSE_BRIDGE	22
FSDKP_NOSE_LEFT_WING	43
FSDKP_NOSE_LEFT_WING_OUTER	45
FSDKP_NOSE_LEFT_WING_LOWER	47
FSDKP_NOSE_RIGHT_WING	44
FSDKP_NOSE_RIGHT_WING_OUTER	46
FSDKP_NOSE_RIGHT_WING_LOWER	48
FSDKP_MOUTH_RIGHT_CORNER	3
FSDKP_MOUTH_LEFT_CORNER	4
FSDKP_MOUTH_TOP	54

FSDKP_MOUTH_TOP_INNER	61
FSDKP_MOUTH_BOTTOM	55
FSDKP_MOUTH_BOTTOM_INNER	64
FSDKP_MOUTH_LEFT_TOP	56
FSDKP_MOUTH_LEFT_TOP_INNER	60
FSDKP_MOUTH_RIGHT_TOP	57
FSDKP_MOUTH_RIGHT_TOP_INNER	62
FSDKP_MOUTH_LEFT_BOTTOM	58
FSDKP_MOUTH_LEFT_BOTTOM_INNER	63
FSDKP_MOUTH_RIGHT_BOTTOM	59
FSDKP_MOUTH_RIGHT_BOTTOM_INNER	65
FSDKP_NASOLABIAL_FOLD_LEFT_UPPER	50
FSDKP_NASOLABIAL_FOLD_LEFT_LOWER	52
FSDKP_NASOLABIAL_FOLD_RIGHT_UPPER	51
FSDKP_NASOLABIAL_FOLD_RIGHT_LOWER	53
FSDKP_CHIN_BOTTOM	11
FSDKP_CHIN_LEFT	9
FSDKP_CHIN_RIGHT	10
FSDKP_FACE_CONTOUR1	7
FSDKP_FACE_CONTOUR2	5
FSDKP_FACE_CONTOUR12	6
FSDKP_FACE_CONTOUR13	8

Tabuľka 1 Zoznam rozoznávajúcich bodov

3.3 Zhrnutie šprintu

Cieľom šprintu bolo implementovať analyzované riešenia z predchádzajúceho šprintu. Počas šprintu sme však identifikovali problémy, kvôli ktorým sme sa rozhodli analyzované riešenia nahradiť. V oblasti spracovania obrazu išlo najmä o použitie projektu Luxand.