

Slovenská technická univerzita
Fakulta informatiky a informačných technológií

Rečové poruchy
Dokumentácia k inžinierskemu dielu

Členovia tímu: Bc. Peter Demčák, Bc. Ondrej Galbavý, Bc. Miroslav Šimek, Bc. Veronika Štrbáková

Vedúci tímu: Ing. Michal Barla, PhD.

Akademický rok: 2013/14

Obsah

1	Úvod	1
2	Šprint 1 - Čučoriedka	3
2.1	Doplniť InstallationID k sessions do databázy	3
2.2	Poskytnutie zoznamu ChildTokenov podľa UserTokenu	4
2.3	Kontrola pred spustením TalkLandu	4
2.4	Oživenie hry Tail Trails v bývalom TalkLande	6
2.5	Ochrana používateľa pred príliš veľkým množstvom offline sessions	8
3	Šprint 2 - Malina	9
3.1	Analýza a návrh TalkLandMedia knižnice	9
3.2	Pridať timestamp do databázy pri vytváraní dieťaťa	10
3.3	Redizajn Speekle webu	11
3.4	Dashboard, prehľadné zobrazovanie štatistík	13
3.5	Spracovanie sessions bez ChildToken na serveri	14
3.6	Pridanie a upravenie dieťaťa priamo z TalkLandu	16
4	Šprint 3 - Egreš	17
4.1	Prehodenie XNA hier do MonoGame	17
4.2	Reset hesla vo webovej aplikácii	17
4.3	Bootstrap TalkLand media knižnice	18
4.4	Zlepšenie mechaniky kúzelného plášťa	19
4.5	Otestovať full instalačku	19
5	Šprint 4 – Jahoda	21
5.1	Doladenie Goldyho v MonoGame	21
5.2	Redizajn Speekle webu s bootstrapom	22
5.3	Pridanie platobného rozhrania s možnosťou použiť externý modul	24
5.4	Prerobiť ukladanie a prehrávanie samplov z .wav na .ogg	24
5.5	Umožnenie priradenia externe nahratej nahrávky k slovu	25
5.6	Generovanie slov do TalkLand hry	25
5.7	Nasadiť dashboard na tímový server	26
5.8	Rozšírenie kontroly chybových stavov v odinštalátore	26
5.9	Zriadiť virtuálny server s OS Windows	27
5.10	Pripraviť IIS pre webovú aplikáciu	27
5.11	Presunúť PHP služby na virtuálny server	28
6	Šprint 5 – Slivka	29

6.1	Integrácia TalkLandMedia knižnice do hry TailTrails.....	29
6.2	Evidencia a zobrazovanie objednávok.....	29
6.3	Vytváranie a prehliadanie párov slov	30
6.4	Doriešiť odinštaláciu TalkLandu	32
7	Zimný semester - Celkový pohľad	33
7.1	TalkLand.....	33
7.2	TalkLand media.....	33
7.3	Goldy	33
7.4	TailTrails.....	34
7.5	Webové rozhranie	34
7.6	API webovej aplikácie pre TalkLand	36
7.6.1	GetUserToken.....	36
7.6.2	HasPaid	38
7.6.3	GetTokens.....	39
7.6.4	DeclInstallationCount	40
7.6.5	CreatePatientTalkLand	40
7.6.6	EditPatientTalkLand.....	42
7.7	Inštalátor	44
7.8	Dashboard	44
7.9	Služby pre TalkLand	45
7.9.1	Autorizačná služba TalkLand	45
7.9.2	Služba pre trial tokeny.....	45
7.9.3	Služba pre zaznamenávanie výnimiek.....	45
7.9.4	Služba pre záznam udalostí	46

1 Úvod

Tento dokument opisuje priebeh vývoja projektu Speekle na predmete Tímový projekt v rámci inžinierskeho štúdia na Fakulte informatiky a informačných technológií STU. Priebežný vývoj stavu projektu je bližšie popísaný vo forme príbehov z jednotlivých šprintov, na ktoré tím prácu vyčlenil.

Víziou projektu Speekle je eliminovať rečové poruchy v populácií. Speekle poskytuje podporu rečovej terapie u detí v predškolskom veku tým, že motivuje logopedické cvičenia prostredníctvom hier.

Medzi ciele riešenia Speekle patrí sprostredkovanie správnych logopedických cvičení používateľom zábavnou a vekovo prislúchajúcou formou. Na toto slúži naša herná platforma TalkLand, ktorej základná verzia už bola vyvinutá v priebehu predchádzajúcej práce tímu, a ktorú chceme ďalej rozvinúť na spoľahlivý softvérový produkt, ktorý podporuje rozličné typy logopedických cvičení – cvičenia fixácie výslovnosti, cvičenia svalov jazyka, cvičenia zvukovej diferenciacie atď. vo forme rôznych špeciálne vyvinutých hier.

Mimo platformy TalkLand je cieľom projektu priniesť logopedické hry aj na tablety alebo mobilné telefóny.

Ďalším cieľom Speekle je integrácia nášho riešenia ako užitočnej pomôcky v rámci procesu rečovej terapie. Rodičom a logopédom chceme preto poskytnúť prehľadný prístup k prehľadu nad postupom terapie dieťaťa, a efektívny spôsob kontroly nad priebehom cvičenia. Na to slúži Speekle webová aplikácia, ktorú chceme urobiť atraktívnou pre používateľov, efektívnou na používanie, a otvorenou pre ďalší rozvoj a rast.

V rámci práce počas zimného semestra bolo našim cieľom dostať projekt do fázy, v ktorej je možné jeho poskytnutie širokej verejnosti. Príbehy v rámci zimného semestra preto zahŕňajú prevažne funkcionality, ktorá zabezpečuje zlepšenie použiteľnosti TalkLandu a webovej aplikácie, vyriešenie bezpečnosti a systému licencií a platieb, a nápravu ďalších existujúcich nedostatkov v našom riešení. Naším druhým cieľom v priebehu zimného semestra bola príprava prvých prototypov hier založených na technológiách, ktoré podporujú expanziu na odlišné platformy, a rozvoj prototypov hier určených pre iné typy cvičení.

2 Šprint 1 - Čučoriedka

2.1 Doplniť InstallationID k sessions do databázy

Analýza:

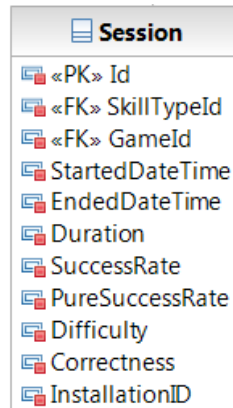
Pre pridanie ID inštalácie k štatistikám z cvičenia sme si zvolili migráciu, ktorá doplní stĺpec k už existujúcej tabuľke Session v databáze a východiskovo nastaví hodnotu na null.

Návrh:

Do databázy je potrebné pridať ID inštalácie k zaznamenávaným štatistikám hier. Pri ukladaní štatistik z cvičení, ktoré sú posielané z TalkLandu, treba uchovávať aj ID inštalácie, pretože do TalkLandu sa plánuje dorobiť anonymný mód, pod ktorým sa dieťa môže hrať v skúšobnej verzii bez prihlásenia sa rodiča do TalkLandu.

Existujúcu tabuľku Session rozširujeme o stĺpec InstallationID, ktorý je typu nvarchar(50), pričom sa do neho môže vkladáť aj null. Nad existujúcou databázou spustíme migráciu databázy, v ktorej tento stĺpec pridáme, pričom východisková hodnota je null.

Návrh databázy je znázornený na obrázku číslo 1.



Column Name	Primary Key	Foreign Key
Id	Yes	No
SkillTypeId	No	Yes
GameId	No	Yes
StartedDateTime	No	No
EndedDateTime	No	No
Duration	No	No
SuccessRate	No	No
PureSuccessRate	No	No
Difficulty	No	No
Correctness	No	No
InstallationID	No	No

Obr. 1 – tabuľka Session

Implementácia:

Pridanie ID inštalácie sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4. Migrácia databázy je zabezpečená objektovo-relačným mapovačom Entity Framework, pričom zdrojový kód migrácie obsahuje 2 metódy:

- ```
public override void Up()
{
 AddColumn("dbo.Session", "InstallationID", c => c.String(maxLength: 50));
}
```
- ```
public override void Down()
{
    DropColumn("dbo.Session", "InstallationID");
}
```

Testovanie

Doprogramovanú funkcionálnosť sme nasadili na server a bola dostupná na stránke: <http://staging.speekle.sk>. Migrácia v databáze nasadenej na Sql Servri 2012 prešla bez

problémov a ku každému záznamu štatistiky z cvičenia sa pridal stĺpec InstallationID východiskovo nastavený na null.

2.2 Poskytnutie zoznamu ChildTokenov podľa UserTokenu

Analýza:

Predchádzajúce riešenie poskytovalo tokeny detí na základe HTTPS requestu Post TalkLandu, pričom sa posielali meno a heslo používateľa. Teraz TalkLand pri požiadavke na vrátenie tokenov detí posiela iba token usera – userToken, na základe ktorého sa dá používateľ jednoznačne identifikovať.

Návrh:

V metóde public ActionResult GetTokens() je parsovanie dát z HTTPS Postu navrhnuté tak, že sa snažíme vybrať token používateľa (userToken) namiesto mena a hesla používateľa (userName, password). Opravené je aj hláška, ktorá sa posielala, ak sa používateľ nenašiel. V predošlej verzii sa posielala v xml súbore správa, že pri požiadavke o tokeny detí bolo zadané zlé meno alebo heslo používateľa. Teraz sa posiela informácia, že používateľ nie je validný.

Implementácia:

Zmenilo sa parsovanie dát z HTTPS Postu pri metóde public ActionResult GetTokens() tak, že hľadáme token používateľa – userToken z dát Postu. Validnosť používateľa overujeme na základe toho, či existuje userToken v databáze zhodný s prijatým userTokenom. Ak áno, odošle sa TalkLandu ako odpoveď zoznam tokenov detí, inak príde TalkLandu nasledujúci xml súbor:

```
<childTokens>
  <result>Invalid user</result>
</childTokens>
```

Testovanie:

Doprogramovanú funkcionálnosť sme nasadili na server a bola dostupná na stránke: <http://staging.speekle.sk>. Požiadavky na tokeny detí posielané HTTPS requestom Post z TalkLandu, pričom sa posielal iba token používateľa – userToken, prešli bez problémov a každému používateľovi vrátili buď zoznam tokenov detí alebo chybu.

2.3 Kontrola pred spustením TalkLandu

Analýza:

Pri kontrole TalkLandu pred spustením jeho hlavného prostredia je potrebné overovať tieto dve základné podmienky

- Či používateľ už zaplatil, a tým pádom už má plnú verziu bez kontroly pred spustením
- Či je skúšobná verzia ešte stále platná

Ďalšou dôležitou súčasťou funkcionality kontroly pred spustením je umožnenie používateľovi prihlásiť sa na základe tohto prihlásenia je nie len identifikovaná jeho platba, ale taktiež sa budú podľa jeho účtu synchronizovať profily detí so serverom.

Väčšina hodnôt pri kontrole pred spustením sa ukladá a číta z registrov pričom kľúčové hodnoty sú rôznymi metódami skryté pred používateľom, aby bolo riziko zmeny používateľom minimálne.

Návrh:

Návrh registrov pre Speekle TalkLand, všetky tieto registre okrem InstallationID, InstallationDate, TrialDays a skrytý InstallationDate sa pri odinštalovaní mažu.

Verejné registre:

- **AccountNumber** : Číslo účtu kam je potrebné poslať peniaze. Tento údaj je len pre informačné účely.
- **Language** : Jazyk aplikácie, možné hodnoty sú sk-SK, en-US.
- **InstallationID** : GUID vygenerované po inštalácii. Jednoznačne identifikuje počítač na ktorom je TalkLand nainštalovaný.
- **InstallationPath** : Cesta kde je TalkLand nainštalovaný.
- **InstallationDate** : Dátum kedy prebehla inštalácia. Tento údaj slúži len na informačné účely, aby bolo možné ihneď po spustení zobrazit' informáciu, že zostáva vám ešte N dní. Zmena tohoto dátumu neovplyvňuje kontrolu pri spustení, keďže na to slúži kryptovaný dátum v skrytých registroch.
- **Price** : Cena, ktorú by mal používateľ zaplatiť. Spolu s AccountNumber a VariableSymbol ide o trojicu informácií potrebnú na zaplatenie za TalkLand.
- **TrialDays** : Doba trvania skúšobnej verzie. Táto hodnota slúži spolu s InstallationDate výlučne na informačné účely z rovnakých dôvodov ako pri InstallationDate. Aby bolo možné jednoducho túto dobu v prípade potreby zmeniť, táto hodnota sa ukladá počas inštalácie.
- **VariableSymbol** : Variabilný symbol na ktorý je potrebné poslať peniaze. Tento údaj je len pre informačné účely.

Skryté registre:

- **TalkLandIsPaid** : Môže obsahovať stavy “zaplatené” a “nezaplatené a skúšobná verzia vypršala”
- **InstallationDate** : Dátum kedy bola skúšobná verzia nainštalovaná. Vytvorí sa len raz na trvalo, keď ešte neexistuje.
- **UserToken** : GUID jednoznačne identifikujúce používateľa. Pomocou neho sa synchronizujú profily detí so serverom, a posiela sa na server pri odinštalovaní, pri kontrole či je zaplatené a tiež sa posiela pri každej session kôli identifikácii sessions keď sa hrá bez profilu dieťaťa.

Implementácia:

Podrobný scenár implementovanej funkcionality kontroly pred spustením TalkLandu:

Časť A - prvé spustenie aplikácie:

- 1) Ak sa v skrytých registroch nachádza InstallationDate pokračuje sa časťou B, inak sa tento InstallationDate vypýta od servera spolu s InstallationDate a TrialDays vo verejných registroch.
- 2) Ak server pošle chybu tak sa dočasne funguje bez skrytého InstallationDate. Verejný InstallationDate bude aktuálny dátum a TrialDays bude 14.
- 3) Pokiaľ nejde internet TalkLand sa ospravedlní, že na prvé spustenie je potrebný internet a zavrie sa.

Časť B - kontrola po spustení:

- 1) Pozrie sa do registrov či používateľ zaplatil. Ak áno spustí sa TalkLand a všetko je v poriadku. Scenár končí.
- 2) Na základe InstallationDate a TrialDays v neskytých registroch sa vypočíta koľko dní ostáva do konca TrialVerzie. Tento údaj sa vypíše v LaunchForm a neskôr potenciálne aj v Prihlasovacom/Informačnom formuláry.
- 3) Ak má v skytých registroch UserToken pošle ho na server s tým, že sa ho spýta, či už nie je náhodou zaplatené. Pokiaľ je, zapíše to do registrov a prepíše správu o zostávajúcich dňoch skúšobnej verzie na úspešnú aktualizáciu na plnú verziu. Scenár končí.
- 4) Pozrie sa do skytých registrov na kryptovaný dátum inštalácie, vyberie ho a pošle na server, či je to v poriadku. Ak sa vráti, že skúšobná verzia je už neplatná tlačidlo na pokračovanie do skúšobnej verzie v prihlasovacom/informačnom formuláry sa zablokuje. Ak je ale skúšobná verzia stále platná a formulár sa podľa jeho príznaku nemá zobrazit', priamo sa spustí aplikácia a scenár končí).

Časť C - prihlasovací/informačný formulár:

- 1) Prihlasovací/Informačný formular sa zobrazuje, pokiaľ je jeho príznak zobrazenia true alebo do konca skúšobnej verzie ostávajú 4 alebo menej dní. V tomto formulári sa používateľ môže prihlásiť a ak je prihlásený vidí svoje platobné údaje.
- 2) Pokiaľ sa používateľ prihlási, a server vráti UserToken, Price, AccountNumber a VariableSymbol, zapíše sa UserToken do skrytých registrov a Price, AccountNumber, VariableSymbol do verejných registrov a znova sa vykoná bod 3) v časti B a formulár zmení vzhľad na ľavej strane z prihlasovacieho na informačný o platbe.
- 3) Pokiaľ používateľ zavrie formular a zaskrtol možnosť "Nabudúce nezobrazovať" (táto možnosť nie je k dispozícii pokiaľ zostáva do konca skúšobnej verzie 4 a menej dní), príznak Prihlasovacieho/Informačného formulára sa nastaví na false.

2.4 Oživenie hry Tail Trails v bývalom TalkLande

Analýza:

Detskí pacienti mávajú často problémy s rozoznávaním odlišných hlások. Jedno z typov cvičení ktoré preto deti vykonávajú u logopéda sú tzv. cvičenia zvukovej diferenciacie.

Cvičenia zvukovej diferenciacie sa vykonávajú nad bázou rôznych slov obsahujúcich rozoznávané hlásky. Tieto cvičenia sa všeobecne delia do niekoľkých kategórií:

- Individuálne (pacient určuje, či individuálne slovo obsahuje hlásku alebo nie)
- Porovnávacie (pacient dostane dve slová a má určiť ktoré obsahuje hlásku a ktoré nie)
 - Odlišné slová (napr. citrón/čučoriedka)
 - Skomoleniny (napr. cibula/čibula)
 - Podobné slová (napr. ocko/očko)

V záujme Speekle ako riešenia na podporu logopedických cvičení je podpora aj takýchto typov cvičení prostredníctvom hier.

V kontexte hernej platformy TalkLand je tiež vhodné, aby rôzne hry mali prístup k tej istej rozširovateľnej množine slov zatriedených podľa hlások, s prislúchajúcim obrazom a zvukovou nahrávkou slova.

Návrh:

Hra TailTrails je hra založená na princípe hry Snake, ktorej herný princíp je založený na konštantne pohybujúcom sa hadovi, ktorý zbiera body na obrazovke, a musí sa vyhýbať narazeniu do okrajov obrazovky a vlastného svojho chvosta. Pri každom zozbieraní bodov sa had predĺži, čím zaberá čoraz väčšiu časť hracej plochy, a hra sa tým kontinuálne sťažuje.

V TailTrails je cieľom okrem zábavy poskytnúť aj cvičenie diferenciacie. Cvičenie je založené na individuálnom cvičení, teda úlohou hráča je identifikovať výskyt konkrétnej hlásky v slove. Vždy po tom, ako sa na obrazovke zobrazí bod, hráč si vypočuje nahrávku slova, a musí identifikovať, či slovo obsahuje hľadanú hlásku. Pokiaľ hráč zoberie správne slovo, získava herné body, zbieraním nesprávnych slov sa hráčovi body strhnú.

Pre uľahčenie pamätania, ktoré hlásky má hráč zbierať, budú jednotlivé hlásky reprezentovať odlišné postavy hadov.

Takáto hra bude vyžadovať prístup k zoznamu slov zatriedených podľa hlások. Navrhované riešenie je poskytnúť hre v rámci TalkLandu rozhranie na prístup k úložisku slov. Volania tried týchto rozhraní by mali byť v tvare:

```
WordGenerator
{
public WordGenerator(Letter mainLetter, List<Letter> similarLetters, bool
includeGarbles);
public Word GetNextWord();
public float CorrectWordProbability {get; set;}
}
```

Vlastníctvo triedy CorrectWordProbability sprístupňuje pravdepodobnosť vygenerovanie slova so správnou hláskou. Táto pravdepodobnosť je riadená hrou a nie samotným rozhraním, pretože neovplyvňuje náročnosť samotného cvičenia, ale priebehu hry.

Implementácia:

Prvý prototyp hry TailTrails vznikol v priebehu minulého roka v rámci analýzy rámca na tvorbu hier XNA Framework.

Tento prototyp hry poskytuje základnú hernú mechaniku, kde hráč ovláda hada na obrazovke šípkami. Hra začína stlačením jednej zo šípok. Na obrazovke sa zjavuje postava bobule, ktorej zozbieranie spôsobí, že had narastie. Stav hracej plochy sa obnoví do počiatočného stavu, keď had narazí do kraja hracej plochy alebo do svojho chvosta.

Had je zložený z kruhových segmentov, pričom kolízia s chvostom znamená kolíziu s kruhovým kolíznym priestorom okolo ktoréhokoľvek segmentu v určitej vzdialenosti od hlavy hada.

```

public class Segment : Sprite
{
    public bool CollideReaction(Snake aSnake)
    {
        if ((Position - aSnake.Position).Length() < (Size.Width / 2
        + aSnake.Size.Width * Snake.COLLISION_RELATIVE_AREA / 2))
            return true;
        return false;
    }

    ...
}

```

Pre ďalší rozvoj tejto hry bolo nutné vyextrahovať kód hry z kódu prvého návrhu TalkLandu, ktorý bol v prvej verzii celistvou hernou aplikáciou. Pre tieto účely boli vyčlenené zo starého projektu triedy hry TailTrails a boli vložené do samostatne spustiteľného XNA projektu.

2.5 Ochrana používateľa pred príliš veľkým množstvom offline sessions

Analýza:

Ak by používateľ mal už TalkLand zaplatený je možné TalkLand používať bez prístupu k Internetu. Týmto spôsobom by mohla nastať situácia, že sessions vytvárané po hraní hier by po dlhom čase zaplnili neprimeranú časť disku, alebo by ho až úplne zahltili. Preto je potrebné maximálny počet lokálne uložených sessions obmedziť.

Návrh:

Pokiaľ sa prekročí maximálny povolený počet sessions vymažú sa najstaršie sessions, konkrétne sessions s najnižším identifikačným číslom.

Implementácia:

V súbore Data.dat je nová položka sessionCountLimit, ktorej predvolená hodnota je 50 a určuje maximálny počet neodoslaných sessions v priečinku Sessions. Mazanie všetkých prebytočných sessions nastáva pred začiatkom nahrávania sessions na server.

Testovanie:

Testovanie prebiehalo nastavením sessionsCountLimit hodnoty na 3, odpojením sa od Internetu, aby sa sessions hromadili lokálne na disku a následne vytvorením troch sessions hraním hry Goldy tri krát po sebe.

Po ich vytvorení sa zaznamenali ich identifikačné čísla a vytvorila sa ešte jedna session hraním hry Goldy. V priečinku sessions sa skontroloval stav sessions, pričom session s najnižším identifikačným číslom mala chýbať a mala pribudnúť nová session s najvyšším identifikačným číslom. Tento korektný stav nastal a testovanie bolo týmto úspešné.

3 Šprint 2 - Malina

3.1 Analýza a návrh TalkLandMedia knižnice

Analýza:

Účelom TalkLand media knižnice je všeobecné poskytovanie všetkých médií spojených s identifikáciou slova a jeho významu pre hry v TalkLande, ktoré budú s týmito slovami pracovať ako napríklad hry na diferenciaciu slov.

Výsledkom analýzy vhodnosti médií na poskytovanie je, že každé slovo je reprezentované tromi druhmi médií a to textom zachytávajúcím jeho písomnú formu, obrázkom zobrazujúcim jeho význam, a nahrávkou jeho výslovnosti.

Každé slovo v TalkLand media knižnici musí patriť pod charakteristickú hlásku, ktorá sa v ňom nachádza a ktorá je z logopedického hľadiska v slove významná vzhľadom na účel na ktorý sa slovo používa pri cvičení.

Slová sú používateľovi poskytované samostatne alebo v pároch.

Pri samostatných slovách sa slová vyberajú určitým algoritmom na základe zadanej charakteristickej hlásky a zadaných nesprávnych ale podobných charakteristických hlások a pravdepodobnosti vybrania slova so správnou charakteristickou hláskou.

Slová poskytované v pároch obsahujú danú dvojicu charakteristických hlások a sú buď obidve zmysluplnými slovami (noc, nos), jedno z nich je skomoleninou (budík, pudík) alebo sú obidve bez významu (klaš, kleš).

Návrh:

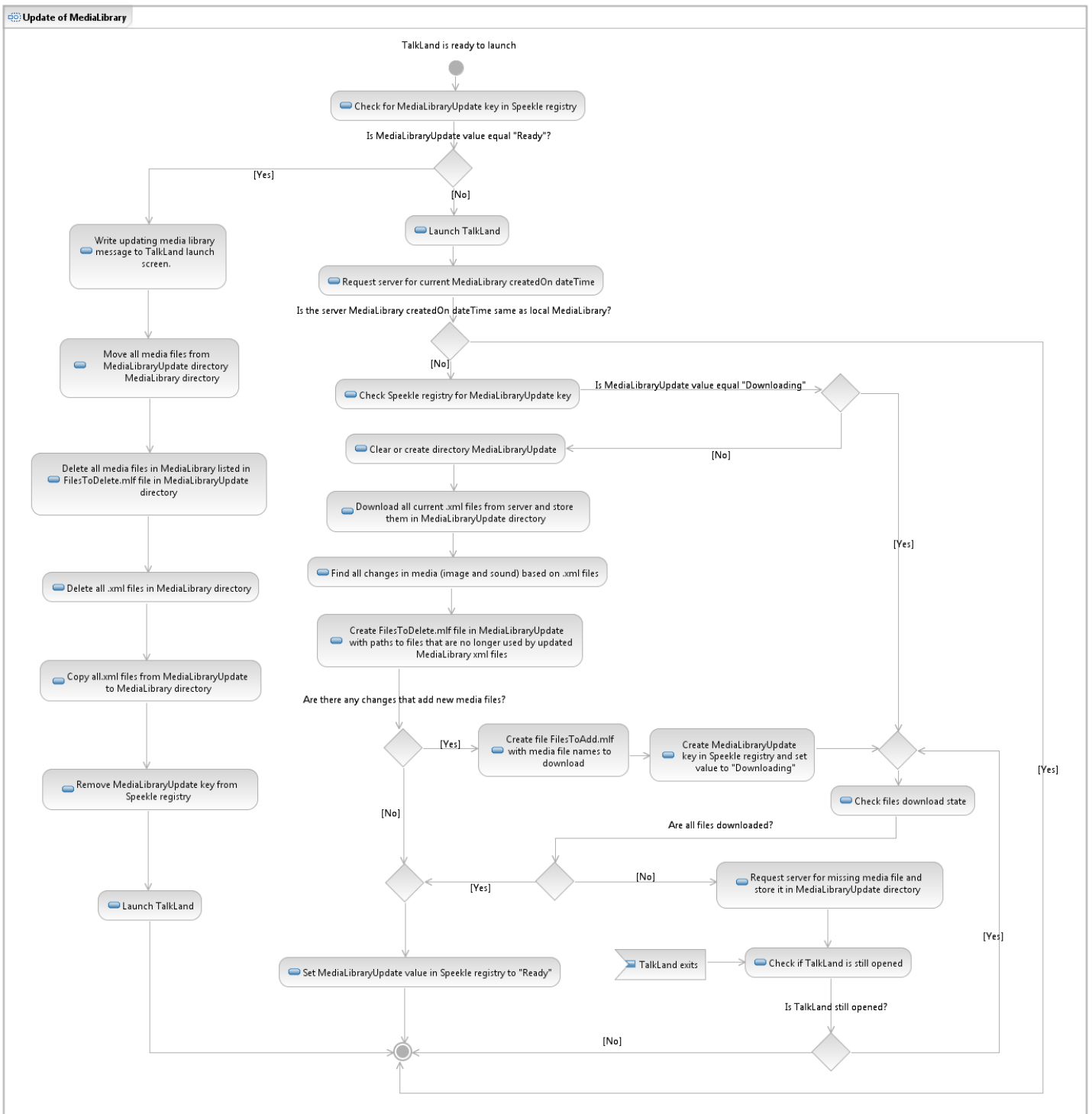
TalkLand media knižnica bude mať slová uložené lokálne v priečinku s názvom MediaLibrary. MediaLibrary pozostáva zo všetkých obrázkov, nahrávok výslovností a z xml súborov pre slová každej charakteristickej hlásky v ktorých sú zachytené naviazania jednotlivých obrázkov a nahrávok na slová a ich rôzne formy.

Povinnou súčasťou každého slova je jeho textová forma a pozícia charakteristickej hlásky v slove. Nepovinne môže mať aj ľubovoľný počet ďalších textových foriem s inou charakteristickou hláskou spolu s pozíciami tejto hlásky v slove, pričom tieto ďalšie textové formy sú skomoleninami hlavnej povinnej textovej formy (bubon, pubon, pupon, bupon).

Každá z textových foriem môže mať k sebe naviazanú jednu nahrávku jej výslovnosti. Všetky textové formy spolu môžu mať k sebe naviazaný ľubovoľný počet obrázkov, ktoré sa budú vyberať pri poskytovaní používateľovi náhodne za účelom väčšej variability a zaujímavosti hier.

Slovo poskytnuté používateľovi TalkLand media knižnicou obsahuje priamo už načítaný jeden vybraný obrázok a je schopné priamo prehrať jeho nahrávku výslovnosti aj s poskytnutím informácie kedy prehrávanie skončilo.

TalkLand media knižnica umožňuje synchronizáciu MediaLibrary zo Speekle serverom pričom táto synchronizácia začne prebiehať po úspešnom skontrolovaní a spustení TalkLandu. Aktualizácia MediaLibrary sa sťahuje do osobitného priečinka a nahradí súčasnú verziu až po kompletnom stiahnutí a ďalšom spustení TalkLandu. Podrobný proces synchronizácie a aktualizácie MediaLibrary je zobrazený diagramom aktivít na obr. 2.



Obr. 2 – Diagram aktivít synchronizácie mediálnej knižnice TalkLandu

3.2 Pridať timestamp do databázy pri vytváraní dieťaťa

Analýza:

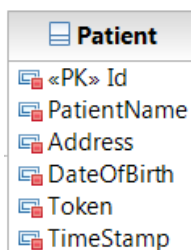
Pre pridanie stĺpca TimeStamp do tabuľky Patient sme si zvolili migráciu, ktorá doplní stĺpec k už existujúcej tabuľke Patient v databáze a východiskovo nastaví hodnotu na null.

Návrh:

Do databázy je potrebné pridať TimeStamp k pacientom. Pri vytváraní dieťaťa sa TimeStamp nastaví na aktuálny dátum.

Existujúcu tabuľku Patient rozširujeme o stĺpec TimeStamp, ktorý je typu dateTime, pričom sa do neho môže vkladať aj null. Nad existujúcou databázou spustíme migráciu databázy, v ktorej tento stĺpec pridáme, pričom východisková hodnota je null.

Návrh databázy je znázornený na obrázku číslo 2.



Obr. 3 – tabuľka Patient

Implementácia:

Pridanie TimeStampu sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4. Migrácia databázy je zabezpečená objektovo-relačným mapovačom Entity Framework, pričom zdrojový kód migrácie obsahuje 2 metódy:

```
1. public override void Up()
    {
        AddColumn("dbo.Patient", "TimeStamp", c => c.DateTime(nullable: true));
    }
1. public override void Down()
    {
        DropColumn("dbo.Session", "InstallationID");
    }
```

Testovanie:

Doprogramovanú funkcionálnosť sme nasadili na server a bola dostupná na stránke: <http://staging.speekle.sk>. Migrácia v databáze nasadenej na Sql Servri 2012 prešla bez problémov a ku každému záznamu o dieťati sa pridal stĺpec TimeStamp východiskovo nastavený na null. Pri vytvorení nového dieťaťa sa TimeStamp nastavil na aktuálny dátum.

3.3 Redizajn Speekle webu

Analýza:

Hlavná stránka speekle.sk obsahuje statickú stránku s kontaktmi na tím, možnosť pridať sa do zoznamu príjemcov newsletter a odkaz na beta verziu webovej aplikácie. Sťahovanie TalkLandu je zatiaľ dostupné len po prihlásení, a samotná webová aplikácia tiež nie je vzhľadovo veľmi príťažlivá. Pre verejnú verziu je preto potrebné, aby sa našlo riešenie ako zlepšiť vzhľad stránky a urobiť samotnú webovú aplikáciu prístupnejšou pre používateľa.

Návrh:

Riešením je spojenie obsahu statickej stránky s webovou aplikáciou, tak aby používateľ získal okamžitý prístup k webovej aplikácii z hlavnej stránky a nestratil pritom funkcionality statickej stránky. Vytvorili sme nasledovné grafické prototypy založené na predchádzajúcom obsahu obrazoviek speekle.sk:



Obr. 4 - Prototyp hlavnej obrazovky



Obr. 5 - Prototyp obrazovky O Speekle

Implementácia:

Redesign Speekle webu sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4.

3.4 Dashboard, prehľadné zobrazovanie štatistík

Analýza:

Produkty Speekle sú už vo verejnej beta verzii ale chýba nástroj na monitorovanie ich používania. Nie sú k dispozícii generované štatistiky o počte jednotlivých typov sedení za deň, ani o inštaláciách. Ani pomer používateľov, ktorí vytvorili najviac sedení.

Databáza webovej aplikácie (MS SQL server) obsahuje informácie o sedeniach a zo štruktúry vyplýva či sú normálne (známe dieťa), neregistrované (známy rodič alebo logopéd) alebo anonymné (známy počítač). Databáza služby pre záznam udalostí (MySQL server) obsahuje informácie o inštaláciách a odinštaláciách.

Návrh:

Vytvorí sa webová aplikácia „Dashboard“, ktorá bude z údajov z databáz generovať prehľady používania produktov Speekle. Typy grafov:

- Počty normálnych, neregistrovaných a anonymných sedení za jednotlivé dni
- Priebehy počtu používateľov a detí
- Počtu inštalácií, aktualizácií a odinštalácií za jednotlivé dni
- Histogram – koľko používateľov vytvorilo určitý počet sedení

Aplikácia sa bude pripájať priamo na MS SQL server webovej aplikácie a MySQL server služby pre záznam udalostí, nebude sa vytvárať API pre dočasný charakter aplikácií.

Implementácia:

Webová aplikácia je vytvorená v jazyku Ruby s využitím rámca Ruby on Rails. Aplikácia má dve stránky, na je prvej sú grafy sedení, počtu používateľov a histogram. Na druhej sú počty inštalácií a iných udalostí. Aplikácia je v princípe jednoduchá a posielá priamo SQL príkazy databázovým serverom.

SQL príkaz pre počty inštalácií za jednotlivé dni:

```
SELECT DATE(date) as date, COUNT(*) as cnt FROM notify_logs WHERE
module='installer' AND action='install' GROUP BY DATE(date)
```

SQL príkaz pre histogram normálnych sedení:

```
SELECT ROUND(count, -1) AS bucket, COUNT(*)AS COUNT
FROM
  (SELECT PatientId as patient_id, count(*) AS count FROM
    (SELECT p.PatientId FROM [Session] AS s
      JOIN [PermissionAndSession] AS pas ON s.ID = pas.SessionID
      JOIN [Permission] AS p ON p.ID = pas.PermissionID
      WHERE p.PatientId IS NOT NULL
      GROUP BY s.id, p.PatientId) as session_patient
    GROUP BY PatientId) as t
  GROUP BY ROUND(count, -1);
```

Grafy sa vykresľujú pomocou JavaScriptovej knižnice Flotr2 a údaje generuje aplikácia v podobe JavaScriptového poľa priamo do kódu stránky.

Webová aplikácia ma problém s perzistentným pripojením na databázové servery, ktoré po určitom čase neaktivity ukončujú spojenie. Aplikácia spojenie vytvára raz pri spustení a už ho ďalej nekontroluje ani nevytvorí znovu pri neúspešnom poslaní príkazu.

3.5 Spracovanie sessions bez ChildToken na serveri

Analýza:

Pri vytváraní plnej verzie TalkLandu, ktorá obsahuje aj skúšobnú verziu, sa pridávajú:

1. anonymný mód – používateľ nie je prihlásený do TalkLandu, na server sa odosiela štatistika bez userTokenu a bez childTokenu, obsahuje iba InstallationID
2. hosťovský mód - používateľ je prihlásený do TalkLandu, ale nemá pre dieťa vytvorený hráčsky profil, na server sa odosiela štatistika bez userTokenu

Teraz už pri spracovaní štatistík je potrebné spracovať aj údaje o cvičení patriace deťom hrajúcim sa pod vyššie spomínanými módmi.

Návrh:

V metóde `public ActionResult Upload()` sa pre štatistiku - Session ukladá aj ID inštalácie – `InstallationID`. Pri nadväzovaní štatistiky na používateľa sa berie v úvahu, či záznam o štatistike obsahuje token používateľa – `userToken`. Ak áno, tak sa overuje, či štatistika obsahuje token dieťaťa – `childToken`. Ak áno, potom sa štatistika naväzuje na používateľa, dieťa na aktuálneho logopéda, ak nejaký existuje. Ak sa v štatistike `childToken` nenachádza, tak cvičenie prebiehalo v hosťovskom móde a naviaže sa iba na používateľa. Ak štatistika neobsahuje ani `userToken` a ani `childToken`, tak sa nenadväzuje na žiadneho používateľa, ale je identifikovaná prostredníctvom `InstallationID`.

Každému používateľovi bol vytvorený záznam v tabuľke `Permission`, ktorý mal `id` dieťaťa – `PatientId` rovné `null`.

Implementácia:

Pridanie `TimeStamp` sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4. Doprogramovalo sa rozhodovanie v metóde `public ActionResult Upload()`, či sa jedná o štatistiku získanú pod hosťovským módom, pod anonymným módom, alebo pod hráčskym profilom dieťaťa. V prípade, že sa jedná o hosťovský mód a anonymný mód, sa kľúčové momenty vyslovovania nenadväzujú na štatistiku.

Migrácia databázy je zabezpečená objektovo-relačným mapovačom Entity Framework, pričom zdrojový kód migrácie obsahuje 2 metódy:

```
2. public override void Up()
    {
        AlterColumn("dbo.Permission", "PatientId", c => c.Int(nullable: true));
        var usersWithRoles = from up in db.UserProfiles
                            join uir in db.UsersInRoles on up.UserId equals
                                uir.UserId
                            select new
                            {
                                UserID = up.UserId,
                                RoleID = uir.RoleId
                            };

        foreach (var item in usersWithRoles)
        {
            Permission per = new Permission();
            per.UserProfileId = item.UserID;
            per.RoleId = item.RoleID;
            per.From = DateTime.Now;
            per.Accept = 1;
            per.Visible = 1;
            db.Permissions.Add(per);

            Sql(string.Format(@"INSERT INTO dbo.Permission (UserProfileId, RoleId,
            ""From"", Visible, Accept) VALUES ('{0}','{1}','{2}','{3}','{4}')"", per.UserProfileId,
            per.RoleId, per.From.ToString("yyyy-MM-dd H:mm:ss"), per.Visible, per.Accept));
        }
    }

2. public override void Down()
    {
        DropColumn("dbo.Permission", "PatientId");
        var usersWithRoles = from up in db.UserProfiles
```

```

        join uir in db.UsersInRoles on up.UserId equals
        uir.UserId
        select new
        {
            UserID = up.UserId,
            RoleID = uir.RoleID
        };

    foreach (var item in usersWithRoles)
    {
        Sql(string.Format("DELETE FROM dbo.Permission WHERE UserProfileId =
        '{0}' AND RoleId = '{1}' AND PatientId is null"), item.UserID, item.RoleID));
    }
}

```

Testovanie:

Doprogramovanú funkcionálnosť sme nasadili na server a bola dostupná na stránke: <http://staging.speekle.sk>. Otestované boli všetky tri možnosti odosielania štatistík, pričom sa tieto štatistiky ukladali do databázy. Pri všetkých štatistikách ukladanie prebehlo korektne a nenastala žiadna chyba.

3.6 Pridanie a upravenie dieťaťa priamo z TalkLandu

Analýza:

V rámci intuitívnosti a prívetivosti používateľského rozhrania je potrebné, aby bolo možné pridávať a upravovať profily detí priamo z aplikácie TalkLand.

Na tento účel je potrebné vytvoriť webovú službu na strane servera, ktorá túto funkcionálnosť sprostredkuje klientovi, čiže aplikácii TalkLand. Taktiež je potrebné rozšíriť webovú službu na získavanie identifikátorov detí podľa identifikátora používateľa, aby vracala nie len identifikátor dieťaťa ale aj jeho meno, dátum narodenia a kód logopéda.

Návrh:

V prípade, že profil dieťaťa ešte nexistuje posiela sa na server tieto údaje: identifikátor používateľa, meno, dátum narodenia a kód logopéda a odpovedá sa OK aj s vytvoreným identifikátorom dieťaťa alebo FAIL s chybovou správou.

V prípade, že profil dieťaťa už existuje posiela sa na server aj identifikátor dieťaťa a server odpovedá len OK alebo FAIL s chybovou správou.

Implementácia:

Pridávanie a upravovanie detí sa v používateľskom rozhraní umožňuje v nastaveniach pod zoznamom profilov detí. Pokiaľ je prihlásený používateľ logopédom pridávanie a upravovanie detí sa mu nezobrazuje, pretože táto funkcionálnosť zatiaľ nie je dostupná ani vo webovom rozhraní.

Pri pridávaní alebo upravovaní detí môžu nastať tieto chyby, ktoré pridanie alebo upravenie znemožnia: Nekorektný formát dátumu, Dátum väčší ako aktuálny dátum, Neexistujúci kód logopéda.

Lokálne meno dieťaťa v aplikácii TalkLand sa zmení spolu so zmenou mena dieťaťa pokiaľ je meno dieťaťa a lokálne meno dieťaťa v aplikácii TalkLand totožné.

4 Šprint 3 - Egreš

4.1 Prehodenie XNA hier do MonoGame

Analýza:

Prototypy hier Goldy a TailTrails vznikli s využitím technológie XNA Framework, čo je voľne dostupný rámec na tvorbu hier. Cieľom Speekle je umožniť ich ďalší rozvoj, zlepšiť ich prístupnosť, a zároveň umožniť ich prenos na tablety, ktoré sú pre detských používateľom najintuitívnejšími zariadeniami na používanie a hranie hier.

XNA Framework však prestal byť ďalej vyvíjaný svojou poskytujúcou spoločnosťou Microsoft. XNA má tiež problémy s kompatibilitou na iných zariadeniach ako tých, ktoré obsahujú operačný systém zo série Microsoft Windows.

Návrh:

MonoGame (*Link: <https://monogame.codeplex.com/>*) je Open Source implementácia XNA 4 Framework, ktorá má za cieľ umožniť prenositeľnosť hier v štruktúre XNA na platformy ako Android, iOS, Linux, alebo herné konzoly. Zároveň zostáva naďalej podporovaný svojou komunitou.

Vďaka takmer totožnému programovaciemu rozhraniu s XNA je možné urobiť relatívne ľahký prechod medzi týmito dvoma prostrediami.

Implementácia:

Aktuálna implementácia MonoGame nie je schopná kompilácie obsahových súborov (písmo, textúra, zvuk) do vyžadovaného formátu XNB, tak ako je tomu u projektov v rámci XNA. Pre vytvorenie funkčného projektu s umožnením jednoduchého prenosu medzi platformami bol sledovaný návod¹.

V rámci MonoGame boli rozmery obrazovky v nastavení grafického zariadenia nastavené na predvolenú hodnotu, zatiaľ čo v XNA odrážali skutočné rozmery obrazovky. Toto spôsobovalo problém so škálovaním, ktorý bol vyriešený nasledovnou modifikáciou konštruktora grafického zariadenia.

```
graphics = new GraphicsDeviceManager(this)
{
    PreferredBackBufferWidth = Screen.PrimaryScreen.Bounds.Width
    PreferredBackBufferHeight = Screen.PrimaryScreen.Bounds.Height
};
```

S dosiaľ neznámych príčin v MonoGame projekte hry Goldy nefunguje zvukový vstup, čo bude vyžadovať hlbší prieskum v nasledujúcej fáze pre plné fungovanie hry v MonoGame.

4.2 Reset hesla vo webovej aplikácii

Analýza:

V prípade, že používateľ zabudne svoje heslo, má možnosť si heslo zresetovať. Dostupných je viacero možností, ako umožniť používateľovi reset hesla:

¹ <http://dementedvice.wordpress.com/2013/06/28/cross-platform-monogame-project-creator-v0-0-1/>

1. Odoslať mu vygenerované heslo na emailovú adresu
2. Zmeniť mu heslo manuálne v databáze a poslať mu ho na emailovú adresu
3. Vytvoriť resetovací link, ktorý sa posiela na emailovú adresu, pričom je používateľovi umožnené nastaviť si nové heslo

V našom riešení sme sa rozhodli pre možnosť číslo 3.

Návrh:

Používateľ v prípade, že zabudne svoje heslo, klikne na hypertextový odkaz „Zabudli ste svoje heslo?“, ktoré ho presmeruje na podstránku, kde zadá svoju emailovú adresu. V prípade, že je správna, tak sa vytvorí 24-hodinový token resetu hesla a používateľovi sa pošle resetovací link na emailovú adresu. Po kliknutí na tento link je používateľ presmerovaný na stránku, kde si v prípade, že nevypršala doba resetovacieho tokenu, môže používateľ zmeniť heslo. Nové heslo sa ukladá do databázy a používateľ je presmerovaný na správu pacientov / detí.

Implementácia:

Reset hesla sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4.

4.3 Bootstrap TalkLand media knižnice

Analýza a návrh TalkLand media knižnice boli vypracované v rámci predošlého šprintu.

Implementácia:

Súčasťou implementácie TalkLand media knižnice je v prvom rade implementácia aplikácie TalkLand media explorer, ktorá umožní, prehľadné zobrazovanie a upravovanie obsahu MediaLibrary.

Funkcionality implementované v tomto šprinte sú:

- Pridávanie, prehliadanie a mazanie balíčkov slov (so spoločnou charakteristickou hláskou)
- Pridávanie, prehliadanie a mazanie slov
- Pridávanie, prehliadanie a mazanie skomolených foriem slov
- Pridávanie, prehliadanie a mazanie obrázkov k slovám
- Narhávajúce a prehrávanie výslovností k formám slov

Zaujímavou funkcionalitou je hlavne nahrávanie a prehrávanie výslovností k formám slov. Používateľ si môže prehľadne nahráť viacero nahrávok slova a ľubovoľnú z nich si prehrať a následne jednu z nich vybrať ako najlepšiu a priradiť ju k danému slovu. Každá nahrávka sa hneď po nahrať spracuje niekoľkými základnými úpravami.

V prvom rade sa z nej vymaže zo začiatku a z konca prebytočné ticho. Následne sa začiatok a koniec stíšia do stratena, aby nebolo počuť puknutie pri prehrávaní, ktoré sa vyskytuje vždy, keď je v nahrávke určitá konštantná hladina šumu. Na záver sa znormálne hlasitosť nahrávky tak, aby priemerná hlasitosť nahrávok ostala približne konzistentná.

Po zvolení najlepšej nahrávky na priradenie slovu a taktiež po priradení nového obrázka k slovu sa toto novo priradené medium premenuje tak, aby jeho názov zodpovedal slovu ku ktorému je priradené a zároveň, aby bol názov unikátny vzhľadom na existujúce, ale aj už vymazané media z MediaLibrary. Toto je potrebné kôli synchronizácii, ktorá na princípe tejto unikátnosti funguje.

Unikátnosť názvov súborov je dosiahnutá tým, že media sa premenujú podľa formátu *charakteristickáHláska-slovo-skomolenina-poradovéČíslo.pripona*, príklady: “b-budík-2.jpg“, “b-bubon-pubon-3.wav“. Skomolenina sa v názve nachádza len ak ide o nahrávku výslovnosti skomoleniny a poradové číslo len ak je väčšie ako jedna. Poradové číslo sa určuje na základe špeciálneho súboru, ktorý obsahuje všetky názvy mediálnych súborov, ktoré sa už niekedy v MediaLibrary nachádzali a k týmto jednotlivým názvom sú priradené ich posledné pridelené poradové čísla.

4.4 Zlepšenie mechaniky kúzelného plášt'a

Analýza:

V rámci hry Goldy slúži pomôcka kúzelného plášt'a hráčovi na dočasné ukrytie sa pred drakom bez vyslovovania cieľenej hlásky. Plášt' trvá po určitú dobu, počas ktorej je hráč pred drakom ukrytý. Pokiaľ sa však plášt' hráčovi minie tesne pred tým ako sa drak zobudí, hráč nemá čas na túto situáciu zareagovať, a môže byť touto pomôckou frustrovaný.

Je nutné prerobiť mechaniku kúzelného plášt'a tak, aby viac zapadala do celkovej mechaniky cyklu zobudzajúceho sa draka.

Návrh:

Výdrž plášt'a nie je závislá od času, ale trvá do časového okamihu, kedy skončí aktuálny cyklus draka.

Implementácia:

Mechanika plášt'a je z implementačného hľadiska zjednodušená, namiesto časovača a implementácie blikania pred ukončením výdrže plášt'a túto mechaniku spravujú iba dve metódy aktivácie a deaktivácie plášt'a.

```
public void WearCloak(Cloak cloak)
{
    invisibleCloakActive = true;
    Invisible = true;
    ResetFrameRectangle();
}

public void DepleteCloak()
{
    if (invisibleCloakActive)
    {
        invisibleCloakActive = false;
        Invisible = false;
        ResetFrameRectangle();
    }
}
```

Deaktivácia plášt'a sa aktivuje pri udalosti uspatia draka.

4.5 Otestovať full instalačku

Analýza:

Chystáme na vypustenie plnú verziu TalkLand, ktorá sa inštaluje na používateľov počítač. Na to slúži inštalátor naprogramovaný v jazyku nástroja NSIS. V zdrojovom kóde inštalátora

došlo k viacerým úpravám a vylepšeniam a preto treba inštalátor otestovať na viacerých rôznych reálne používaných počítačoch.

Návrh:

Každý člen tímu otestuje pripravený inštalátor na počítačoch, ktoré ma k dispozícii a ešte na nich nebol nainštalovaný TalkLand.

Výsledok testovania:

Inštalátor bol otestovaný na piatich počítačoch a úspešne nainštaloval potrebné závislosti. Identifikovali sme nedostatky odinštalátora a jeho integrácie so samotným TalkLandom:

- Ak server pri deaktivácii TalkLandu zlyhá (chyba 500), odinštalátor to interpretuje ako nefunkčné pripojenie na Internet
- Ak používateľ nebol prihlásený, tak odinštalátor situáciu interpretuje ako nefunkčné pripojenie na Internet
- Dialóg na vynútené odinštalovanie ma nedostatky v použiteľnosti.
- Ak TalkLand bol spustený v čase odinštalovania, tak spadne a určité súbory nebudú vymazané

Prvé dva body boli už vyriešené na strane TalkLandu rozšírením návratových kódov, ktoré bude odinštalátor správne interpretovať.

5 Šprint 4 – Jahoda

5.1 Doladenie Goldyho v MonoGame

Analýza:

Po prenesení hry Goldy z rámca XNA do rámca MonoGame začala hra preukazovať neočakávané správanie. Predchádzajúca verzia hry bola ovládaná v niektorých fázach zvukom, avšak verzia hry v MonoGame neakceptovala žiaden zvukový vstup, čím vážne blokovala celkový účel hry. Ďalej bolo v hre Goldy odpozorované, že po určitej dobe od spustenia (ale nie vždy, a nebolo zjavné prečo) sa v hre vyskytla výnimka s nasledujúcim znením:

Managed Debugging Assistant 'ContextSwitchDeadlock' has detected a problem in 'C:\Users\Peter\Documents\Visual Studio 2012\Projects\Goldy\Goldy_WindowsGL\bin\Debug\Goldy_WindowsGL.vshost.exe'.

Additional Information: The CLR has been unable to transition from COM context 0x5a8310 to COM context 0x5a8480 for 60 seconds. The thread that owns the destination context/apartment is most likely either doing a non pumping wait or processing a very long running operation without pumping Windows messages. This situation generally has a negative performance impact and may even lead to the application becoming non responsive or memory usage accumulating continually over time. To avoid this problem, all single threaded apartment (STA) threads should use pumping wait primitives (such as CoWaitForMultipleHandles) and routinely pump messages during long running operations.

If there is a handler for this exception, the program may be safely continued.

Ani po prenesení hry TailTrails do MonoGame, ani pri vytvorení úplne nového MonoGame projektu sa totožná výnimka v hrách nevyskytovala. Účelom tohto rozšírenia bolo preto počas šprintu odstrániť z MonoGame hry Goldy tieto závažné nedostatky. Podstata tejto úlohy bola v značnej miere analytická, pretože jej významnou súčasťou bolo zistiť príčinu daného problému.

Po debugovaní spojených projektov Goldy a SoundAnalyzer bola odhalená príčina neregistrovaného zvukového vstupu. Trieda WaveIn z knižnice NAudio vykonávala volania o prístupnosti zvukových údajov cez mechanizmus message looping, ktorý je využívaný v rámci paralelizmu vo Windows.Forms na posielanie správ medzi vláknami. Keďže MonoGame message looping nepodporuje, správy neboli obsluhované a zvukový vstup nebol knižnicou registrovaný.

Z tohto jadra pochádzal aj druhý problém hry Goldy v MonoGame. Výnimka ContextSwitchDeadlock bola vyvolaná vláknom, ktoré pumpovalo nové správy ktoré následne neboli ani po vypršaní časovača obslužené.

Ani po odstránení všetkých významných komponentov hry však nebola problémová výnimka odstránená. Pri prerušení hry pri výskyte výnimky bolo vlákno vyvolávajúce výnimku označené za nedostupný a nebežiaci kód. Preto bolo problematické identifikovať zdroj odosielaných správ.

Návrh:

Po prieskume existujúcich riešení podobných problémoch sa objavili dva spôsoby riešenia. Volanie metódy `Application.DoEvents` vyvoláva obsluhu pumpovaných správ manuálne počas behu programu. Druhou alternatívou je problém prakticky ignorovať, nakoľko podľa viacerých zdrojov je táto výnimka len varovná, objavuje sa len v debuggeri, ale nie už počas samotného riadneho behu programu. Po zistení, že metóda `Application.DoEvents` spôsobuje narušenie vstupu z klávesnice v rámci MonoGame hry, sme zvolili druhý spôsob riešenia.

Implementácia:

V rámci knižnice `SoundAnalyzer` bola trieda `WaveIn` nahradená triedou `WaveInEvent`, ktorá nie je závislá na modeli `message looping`, a preto má všeobecnejšie možnosti využitia v rôznych typoch aplikácií. Vo IDE `Visual Studio` bola zavedená výnimka pre výnimku typu `'ContextSwitchDeadlock'` a tým bol eliminovaný jej výskyt počas vývoja hry.

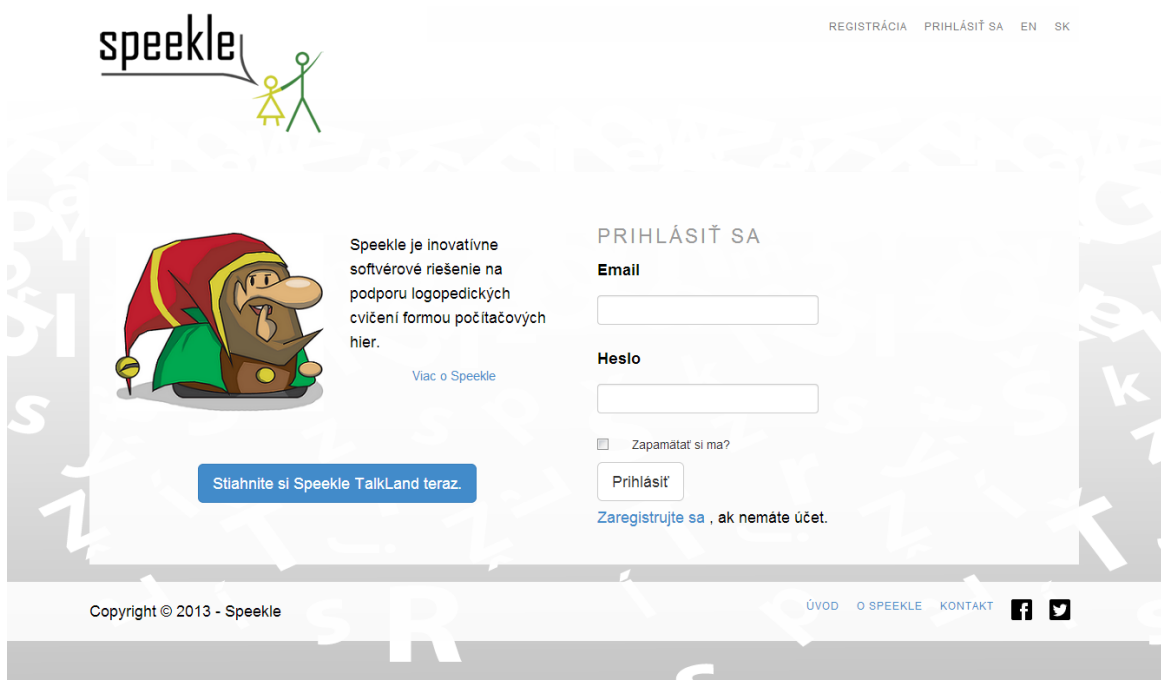
5.2 Redizajn Speekle webu s bootstrapom

Analýza:

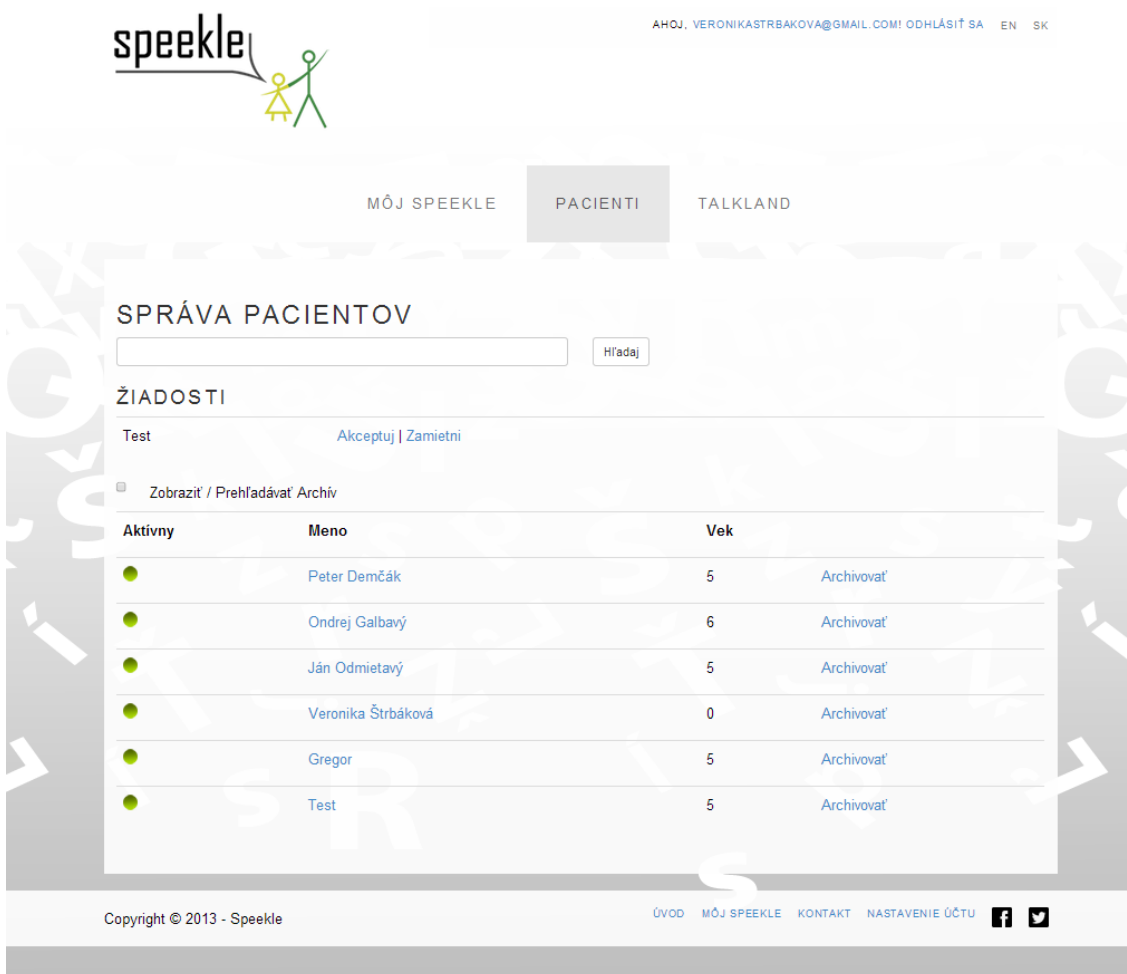
Po prvom redizajne webovej aplikácie v predošlom šprinte sa rozhodlo, že sa použije v riešení `bootstrap`, aby aplikácia nabrala moderný vzhľad a bola prispôbena pre všetky typy zariadení (tablet, smartfón, ...). To znamená, zmazať vlastné štýly webovej aplikácie a použiť štýly `bootstrapu`, pričom sa môže pridať aj vlastné štýlovanie, i keď len veľmi opatrne, aby sa neporušil štýl stránky pri zmenšovaní alebo zväčšovaní zobrazovacej plochy. S tým sa spája aj opravenie všetkých pohľadov (*views*), pričom treba odstrániť všetky staré `div` elementy a použiť typické `bootstrapovské` rozloženie a opraviť všetky formuláre a tabuľky. Na základe predošlého redizajnu sa navrhli aj úpravy úvodnej stránky a celkovo prerozdelenie textu medzi jednotlivé pohľady.

Návrh:

Riešením je prerozdelenie obsahu medzi úvodnú stránku a stránky o `Speekle` a o `TalkLande`, tak aby používateľ získal okamžitý prístup k webovej aplikácii z hlavnej stránky a nestratil pritom funkcionality úvodnej informatívnej stránky. Vytvorili sme nasledovné grafické prototypy založené na predchádzajúcom obsahu obrazoviek:



Obr. 6 – Úvodná stránka Speekle



Obr. 7 – Správa pacientov

Implementácia:

Redizajn Speekle webu sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4.

5.3 Pridanie platobného rozhrania s možnosťou použiť externý modul

Analýza:

Aby sme umožnili používateľom platiť priamo cez naše webové rozhranie, bolo potrebné pridať externý modul do nášho rozhrania a doimplementovať komunikáciu s platobnou bránou. Pre tieto potreby sme zanalyzovali možnosti, ktoré by sme mohli použiť a našli sme 2 možné riešenia:

1. TrustPay
2. Pay24

Nakoniec sme si zvolili externý modul TrustPay, pretože má pokrytých viac bánk vo viacerých krajinách a celkom prehľadnú dokumentáciu k implementácii. Ponúkalo nám riešenie nie len pre online platby a offline platby, ale aj pre platbu kreditnou kartou.

Návrh:

Používateľ v prípade, že chce kúpiť licenciu TalkLandu, klikne na hypertextový odkaz „Kúpiť“, ktorý ho presmeruje na stránku platobnej brány s tým, že sa odosiela Https request s parametrami – AID (číslo účtu), AMT (suma), SIG (podpis, ktorý nás identifikuje), a REF (unikátne číslo objednávky). Tam si vyberie formu platby a zaplatí. Platobná brána ho po zvolenej akcii presmeruje opäť na naše rozhranie, pričom sa zobrazí príslušná stránka zobrazujúca úspech platby, zrušenie platby, alebo ukazujúca chybu, ktorá nastala pri pokuse o platbu. Okrem toho príde aj notifikácia prostredníctvom http protokolu, kde sa nachádza zrealizovaná platba a ďalšie údaje potrebné k ďalšiemu postupu, ako je napríklad zvýšenie počtu inštalácií, ak nikde nenastala chyba, alebo ak sú údaje správne.

Implementácia:

Pridanie platobného rozhrania sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4.

Testovanie:

Pre účely testovania nám boli vygenerované testovacie url zo strany TrustPay, kde sme simulovali platbu a rôzne ďalšie situácie, ktoré mohli nastať, ako napríklad zamietnutie platby.

5.4 Prerobiť ukladanie a prehrávanie smplov z .wav na .ogg

Analýza:

Spomedzi komprimovaných formátov zvuku je po analýze preferovaný .ogg pred známejším formátom.mp3, keďže pri predaných kópiách hry nad 5000, by sme museli začať platiť poplatky za každý predaný kus a prerábanie mediálnej knižnice a systému neskôr by bolo zbytočne náročné.

Návrh:

Dočasné nahrávky sa budú naďalej nahrávať aj prehrávať vo formáte .wav, až po zvolení a potvrdení nahrávky ako výslovnosti slova sa nahrávka konvertuje do formátu .ogg a v tejto

forme sa presunie do MediaLibrary s príslušným názvom podľa slova ku ktorému bola nahrávka nahratá.

Implementácia:

Konverzia je implementovaná pomocou natívnej knižnice libZPlay, a prehrávanie je implementované pomocou .NET knižnice csvorbis z dôvodu GNU GPL licencie libZPlay, ktorá by nás zaviazala k zverejneniu nášho kódu verejnosti v prípade potreby.

5.5 Umožnenie priradenia externe nahratej nahrávky k slovu

Analýza: V prípade, že používateľ uprednostní iné nahrávacie prostredie, alebo už existuje databáza nahratých slov, ktorú chceme použiť, je potrebné umožniť priradenie existujúcej nahrávky k slovu.

Návrh: V aplikácii bude možné priradiť externe nahratú nahrávku vo formátoch .wav, .mp3 a .ogg k jednotlivým formám slov. Súčasťou priradenia je aj konverzia nahrávky do jednotného formátu .ogg

Implementácia: K jednotlivým formám slov pribudla okrem nahrávania, prehrávania a mazania možnosť pridávania externe nahratej nahrávky.

Z dôvodu potreby rovnakej funkcionality konverzie zvukového súboru do formátu .ogg ako pri nahrávaní a potvrdení nahratej nahrávky sa presunula táto funkcionality na vyšeobecnejšiu úroveň a bola znovupoužitá.

5.6 Generovanie slov do TalkLand hry

Analýza:

Vygenerované slová by mali byť čo najviac intuitívne pre používateľa, keďže budú potenciálne v budúcnosti používané aj treťou stranou. Z tohto dôvodu vygenerované slovo bude v sebe obsahovať text, načítaný obrázok, možnosť prehrať výslovnosť aj s udalosťou kedy prehrávanie skončilo, cestu k obrázku na disku a cestu k výslovnosti na disku.

Vďaka poskytnutiu cesty k médiám sa pokryjú prípady, kedy by používateľ nebol schopný s technológiou, ktorú používa využiť média v takej forme ako sú knižnicou poskytované.

Návrh:

Prípady použitia generátora slov sú nasledovné. Generujú sa slová vždy s určitou charakteristickou hláskou, k nim je možné určiť zoznam podobných charakteristických hlások, ktoré sa vygenerujú s určitou pravdepodobnosťou $(1 - p)$, kde p je pravdepodobnosť vygenerovania hlavnej charakteristickej hlásky.

Generovanie môže byť v oboch prípadoch ešte bližšie špecifikované na pozíciu charakteristickej hlásky v slove (na začiatku, v strede, na konci a ich kombinácie).

Implementácia:

Pri implementácii bolo najzaujímavejšie určenie vzorca pravdepodobnosti výberu slova. Jednotlivé potenciálne slová na vybratie si v sebe uchovávajú koľko výberov už neboli vybraté. Táto hodnota postupne zvyšuje pravdepodobnosť, že dané slovo bude možné vybrať.

Pravdepodobnosť umožnenia výberu slova je založená na rastúcej funkcii arc tangens. Vďaka tejto funkcii je pravdepodobnosť na začiatku veľmi nízka. Následne začne pomerne prudko rásť, až sa dostane na vysokú hodnotu, kde sa zastaví a ďalej pokračuje znovu veľmi pomaly.

Testovanie:

Za účelom testovania bola vytvorená aplikácia TalkLand media tester, ktorá poskytuje vygenerovanú trojicu text, obrázok a výslovnosť. Funkcionalita aplikácie spočíva len v generovaní ďalšieho slova pričom sa text, obrázok a výslovnosť menia podľa ďalšieho vygenerovaného slova.

Úspešne sa otestoval klasický scenár aj špeciálne prípady ako generovanie z množiny nula, jedného a dvoch slov.

Taktiež sa vyhodnotila vhodnosť funkcie umožnenia výberu slova, tým, že by prípad vygenerovania rovnakého slova iba s jedným alebo dvomi inými slovami medzi, mal byť pri množine 10 slov nulový alebo veľmi výnimočný. Táto podmienka bola tiež splnená.

5.7 Nasadiť dashboard na tímový server

Analýza:

Aby mohol k dashboard pristupovať ktorýkoľvek člen tímu, bez jeho lokálnej inštalácie, je potrebné umiestniť ho na tímový server. Ruby on Rails aplikácie sa v produkčnom prostredí spúšťajú prostredníctvom servera Passenger, ktorý sa používa s webovým serverom Apache alebo Nginx.

Tímový server už používa Nginx, avšak v podobe zo systémového repozitára ku ktorej nie je možné pripojiť Passenger. Zároveň je potrebné zachovať funkčnosť stránky tímu.

Návrh:

Nahradiť Nginx z repozitára za Nginx skompilovaný s podporou Passenger. Vyskúšať primitívnu Ruby on Rails aplikáciu, potom spojzdnit' Dashboard.

Implementácia:

Balíček Nginx z repozitára bol odinštalovaný. Používateľ nginx bol odobraný a vytvorený nanovo s domovským adresárom /home/nginx. Bol nainštalovaný lokálny RVM, Ruby 2.0, Bundler, Passenger a Jekyll.

Nginx bol nainštalovaná do /opt/nginx. Testovacia RoR aplikácia bola umiestená do /home/nginx/ror/nginx_test. Stránka tímu bola presunutá do /home/nginx/html/tp-webpage. Východzia konfigurácia nginx bola upravená, aby poskytovala RoR aplikácie prostredníctvom Passenger a statickú tímovú stránku.

Dashboard je umiestnený v /home/nginx/ror/dashboard a je možné aktualizovať ho z BitBucket git repozitára prostredníctvom deployment kľúča.

5.8 Rozšírenie kontroly chybových stavov v odinštalátore

Analýza:

Odinštalátor sa pokúša o deaktiváciu TalkLand, ktorý však vracal len jeden chybový kód pre akúkoľvek chybu, vrátane prípadu, keď TalkLand nebol aktivovaný a deaktivácia nebola potrebná.

Chybové kódy, ktoré vracia Talkland, boli rozšírené na nasledovné:

- -100: NoUserToken - TalkLand nie je aktivovaný
- -101: DecrementationOperationFailed - zlyhala deaktivácia na strane servera

- -200: NoInternetConnection - zlyhalo pripojenie na server
- -201: InternalServerError - zlyhal server

Návrh:

Rozšíriť odinštalátor o kontrolu nových chybových kódov a príslušne na ne reagovať. V prípade zlyhania pripojenia na server, odinštalátor oznámi používateľovi potrebu pripojenia na internet.

V prípade zlyhania servera odinštalátor oznámi používateľovi potrebu kontaktovať nás. V každom prípade odinštalátor umožní používateľovi TalkLand aj tak odinštalovať, ale upozorní ho na stratu jednej inštalácie.

Implementácia:

Odinštalátor overuje chybový kód, ktorý vráti TalkLand a príslušne naň reaguje dialógovými oknami.

Testovanie:

Odinštalátor bol otestovaný s odpojeným a pripojením na sieť a s lokálnym serverom, ktorý generoval chybné odpovede.

5.9 Zriadiť virtuálny server s OS Windows

Analýza:

Server Ichiban nemôžeme použiť, pretože využíva akademické licencie operačného systému. Produkčnú verziu webovej aplikácie spustíme na vlastnom virtuálnom serveri s operačným systémom Windows, ktorý bude možné použiť na komerčné účely. Tento server bude treba nakonfigurovať.

Návrh:

Využijeme IaaS platformu Azure, ktorú máme k dispozícii prostredníctvom programu BizSpark.

Implementácia:

Vytvorila sa inštancia VPS na Azure s Windows Server 2012 R2 Datacenter Edition s parametrami: 1 jadro, 1.75 GB. Vytvorili sa potrební používatelia. Hostname servera je fry.cloudapp.net, prihlasovacie údaje na administrátorské konto je uložené v dropboxe.

K serveru je možné pripojiť sa prostredníctvom vzdialenej plochy na neštandardnom porte. Firewall na Azure má povolené nasledujúce služby: HTTP, HTTPS, WebDeploy, FTP, RDP.

5.10 Pripraviť IIS pre webovú aplikáciu

Analýza:

Aby sme mohli nasadiť webovú aplikáciu na virtuálny server, treba nainštalovať a nakonfigurovať webový server IIS a databázový server.

Návrh:

Nainštalovať IIS, WebDeploy, SQL Server 2012 Express a SQL Management Studio. IIS bude odskúšané primitívnou ASP.NET MVC aplikáciou.

Implementácia:

Bol nainštalovaný a nakonfigurovaný IIS s podporou ASP.NET MVC. Nakonfigurovaný WebDeploy umožňuje priame nasadenie webovej aplikácie prostredníctvom Visual Studio. Bol nainštalovaný SQL Server 2012 Express a SQL Management Studio. Testovacia ASP.NET MVC aplikácia je nasadená na <http://fry.cloudapp.net/>.

Testovanie:

Testovacia ASP.NET MVC aplikácia bola nasadená prostredníctvom WebDeploy z Visual Studio. Testovacia aplikácia sa v prehliadači načíta. SQL Management Studio sa dokáže pripojiť na SQL Server.

5.11 Presunúť PHP služby na virtuálny server

Analýza:

Niektoré služby, medzi nimi služba pre autorizáciu spustenia beta verzie, sú nasadené na webhostingu, ktorému končí platnosť. Služby je nutné presunúť a zachovať funkčnosť na ich doménových adresách.

Zoznam služieb:

- Autorizačná služba TalkLand
- Služba pre trial tokeny
- Služba pre zaznamenávanie výnimiek
- Služba pre záznam udalostí

Služby su naprogramované v jazyku PHP a využívajú databázový server MySQL.

Návrh:

Služby sa presunú na tímový server. Na tímový server sa nainštaluje PHP-FPM na spúšťanie PHP skriptov a bude pridaný do konfigurácie Nginx. Služby sa budú nasadzovať z repozitárov na BitBucket.

Služby budú reagovať na staré adresy a aj na nové na subdoménach speckle.sk.

Implementácia:

Na tímový server bol nainštalovaný PHP-FPM, kvôli ktorému bol dočasne zapnutý permissívny režim SELinux. Služby sa nasadzujú z BitBucket repozitárov prostredníctvom nasadzovacieho prístupového kľúča do oddelených adresárov v /home/nginx/html.

Bol nainštalovaný a nakonfigurovaný MariaDB server kompatibilný s MySQL. Zo starého webhostingu boli skopírované databázy.

Priradené subdomény pre služby:

- Autorizačná služba TalkLand: tlauth1.speckle.sk
- Služba pre trial tokeny: tltt1.speckle.sk
- Služba pre zaznamenávanie výnimiek: tllog1.speckle.sk
- Služba pre záznam udalostí: tlnot.speckle.sk

Testovanie:

Funkčnosť služieb bola ručne odskúšaná testovacími vstupmi, kontrolou výstupov a kontrolou obsahu databáz. Autorizačná služba TalkLand prechádza svojim kontrolovacím skriptom.

6 Šprint 5 – Slivka

6.1 Integrácia TalkLandMedia knižnice do hry TailTrails

Analýza:

Účelom hry TailTrails je tréning diferenciacie odlišných hlások. Hra vo svojej aktuálnej podobe funguje len ako klasická hra Snake. Je potrebné, aby hráč hrajúci hru mal možnosť vybrať si, ktorú hlásku má rozoznávať, a pri jej rozoznávaní treba hráčovi ďalej prehrať zvukovú nahrávku slova, ktoré obsahuje buď danú hlásku alebo hlásku jej podobnú. Na tieto účely možno do hry zintegrovat' už rozpracovanú knižnicu TalkLandMedia.

Návrh:

Rozličné hlásky v rámci TailTrails reprezentujú odlišné postavičky, preto si hráč na začiatku vyberie jednu z postavičiek. Aktuálna verzia obsahuje obrázkovú reprezentáciu postavičiek hadíka(hlásky s) a vláčika(hlásky š). Hráč si na začiatku vyberie jednu z týchto dvoch postavičiek, a následne sa spustí hlavná časť hry, pri ktorej hráč danú postavičku ovláda na obrazovke. Na obrazovke sa zobrazujú bobule, ktoré teraz zároveň so svojim objavením spustia zvukovú nahrávku slova s danou (alebo podobnou nesprávnou hláskou). Správnosť zbieraných bobúl' zaliat' nie je v rámci hry vyhodnocovaná.

Implementácia:

Na reprezentáciu postavičiek pri ich výbere boli použité mierne upravené existujúce obrázky postavičiek hadíka a vláčika. Kód bol oproti predchádzajúcej verzii prečistený a upravovaný, aby boli odstránené nepotrebné/nehodné časti, ktoré vznikli počas učenia sa technológie rámca XNA. Knižnica TalkLandMedia bola do hry úspešne zintegrovaná bez významných ťažkostí.

Poznámka: Do budúcnosti bude potrebné, aby knižnica umožnila okamžité ukončenie prehrávaného slova.

6.2 Evidencia a zobrazovanie objednávok

Analýza:

S možnosťou platiť za TalkLand priamo v našom webovom rozhraní pribúda potreba evidovania si objednávok, ktoré už používateľ zrealizoval. Objednávka môže byť vytvorená pre platbu kreditnou kartou, alebo pre platbu online / offline bankingom. Údaje objednávky sú skoro také isté, ale čo sa týka platby kartou, tak sa ukladajú aj údaje o karte ako identifikačné číslo karty, alebo dátum expirácie karty.

Návrh:

Navrhli sme si spoločnú tabuľku objednávok aj pre kreditné karty a aj pre platby online / offline, ktorá je v súlade s požiadavkami TrustPay definovanými v dokumentácii:

```
[Table("Order")]
public class Order
{
    [Key]
    [DatabaseGeneratedAttribute(DatabaseGeneratedOption.Identity)]
    public int Id { get; set; }
    [Required]
    public int UserId { get; set; }
    [Required]
```



```

[StringLength(10)]
public string REF { get; set; }
[Required]
[StringLength(16)]
public string AMT { get; set; }
[Required]
[StringLength(3)]
public string CUR { get; set; }
[Required]
[StringLength(64)]
public string SIG { get; set; }
[StringLength(4)]
public string RES { get; set; }

[StringLength(4)]
public string TYP { get; set; }
[StringLength(4)]
public string RES_NOT { get; set; }
[StringLength(10)]
public string TID { get; set; }
[StringLength(10)]
public string OID { get; set; }
[StringLength(1)]
public string TSS { get; set; }
[StringLength(16)]
public string AMT_NOT { get; set; }

[StringLength(36)]
public string CardID { get; set; }
[StringLength(19)]
public string CardMask { get; set; }
[StringLength(4)]
public string CardExp { get; set; }
[StringLength(7)]
public string AuthNumber { get; set; }
[StringLength(1)]
public string CardRecTxSec { get; set; }
[StringLength(128)]
public string AcqResId { get; set; }
[StringLength(64)]
public string SIG2 { get; set; }
}

```

Implementácia:

Pridanie objednávok sme programovali vo vývojárskom prostredí Microsoft Visual Studio 2012 Ultimate s využitím technológie ASP.NET MVC 4, pričom sme sa riadili pokynmi v dokumentácii od TrustPay.

Testovanie:

Pre účely testovania nám boli vygenerované testovacie url zo strany TrustPay, kde sme simulovali platbu a rôzne ďalšie situácie, ktoré mohli nastať, ako napríklad zamietnutie platby, pričom sme sledovali vytváranie a napĺňanie objednávok v databáze.

6.3 Vytváranie a prehliadanie párov slov

Analýza:

Na prácu s pámi slov je potrebné prerobiť systém ukladania obrázkov a výslovností hlavných slov. Doteraz pri rovnakých slovách s rozličnými charakteristickými hláskami (napríklad c-

noc a n-noc) bolo potrebné znovu priradiť obrázky a výslovnosti ako by to bolo pri novom slove, pretože pri predošlej analýze bol tento prípad vyhodnotený ako výnimočný.

Na základe aktuálnej analýzy je však jednoznačné, že tento prístup je veľmi nepraktický, pretože takéto prípady sa vyskytujú oveľa častejšie ako výnimočne. Z tohto dôvodu sa systém ukladania obrázkov a výslovností prerobí tak, aby rovnaké slová s rôznymi charakteristickými hláskami pristupovali k rovnakému obsahu.

Návrh:

Rovnaký obsah pre rovnaké slová dosiahneme mapovaním obsahu cez spoločný xml súbor, na ktorý sa dané slová odkazujú. Názov tohto xml súboru budú slová o ktoré sa jedná. Vďaka tomuto nie je potrebné vyhľadávať, či dané slovo už existuje, pretože vieme, že daný súbor už existuje alebo neexistuje.

Súbor mapujúci obsah v sebe obsahuje aj počet referencií (na slová), ktoré sa na neho odkazujú. Ak tento počet klesne na 0 (vymazalo sa posledné slovo odkazujúce na daný súbor), vymaže sa mapujúci súbor spolu s obsahom, ktorý mapoval.

Samotné vytváranie párov slov bude umožnené s priamym vytváraním slov v prípade, že dané slová s danou charakteristickou hláskou neexistujú. Páry slov sa budú na seba odkazovať pomocou elementu reference v package xml súbore. Referencia obsahuje unikátne id referencie na odstránenie duplikátov, keďže každé z párov slov sa odkazuje na to druhé slovo. Ďalej obsahuje písmeno a jeho pozíciu v slove a samotné slovo na ktoré sa slovo odkazuje.

Implementácia:

Pri vytváraní párov slov je implementované automatické dopĺňovanie charakteristickej hlásky a jej pozície v slovách, ktoré funguje za predpokladu, že sa slová líšia iba v jednej hláske. Táto funkcionálna veľmi zefektívňuje potenciálne hromadné pridávanie párov slov.

Pri určení slov na vytvorenie páru sa tiež prehľadne zobrazuje, či dané slovo s danou charakteristickou hláskou už existuje alebo ešte nie. Pokiaľ nie, slová sa automaticky vytvoria.

Testovanie:

Pri testovaní boli pokryté všetky základné prípady na otestovanie nasledovným scenárom.

- Vytvorenie a pridelenie obsahu slovu noc pre hlásku c
- Vytvorenie páru slov noc a nos
- Skontrolovanie, či bolo automaticky vytvorené slovo nos
- Vytvorenie páru slov noc a moc, skontrolovanie, či vytvorené slovo noc pre hlásku n má totožný obsah so slovom noc pre hlásku c
- Vymazanie slova noc pre hlásku n
- Skontrolovanie, či sa zachoval obsah slova noc pre hlásku c
- Skontrolovanie, či sa vymazalo spárovanie so slovami noc pre hlásku n a moc
- Vymazanie slova noc pre hlásku c
- Skontrolovanie, či sa vymazal obsah aj mapovanie obsahu pre slovo noc

6.4 Doriešiť odinštaláciu TalkLandu

Analýza:

Odinštalátor maže z registrov len inštalačnú cestu, TalkLand maže takmer všetky registre. Avšak v určitých prípadoch TalkLand registre nemaže a tie ostanú v systéme aj po odinštalácii.

Návrh:

Určiť za ktoré registre je zodpovedný inštalátor a TalkLand.

Implementácia:

Odinštalátor maže všetky verejné registre, okrem tých, ktoré môžu ostať pri neúspešnej deaktivácii alebo sú potrebné pre zachovanie funkčnosti trial verzie. Odinštalátor maže: Language, InstallationPath, AccountNumber, Price, VariableSymbol. TalkLand je zodpovedný za odstránenie IncompleteDeactivation a skrytých registrov.

Testovanie:

Odinštalátor a TalkLand bol otestovaný na prípadoch:

- Aktivovaný TalkLand
- Talkland v trial móde bez prihlásenia
- Aktivovaný TalkLand bez pripojenia na internet

7 Zimný semester - Celkový pohľad

7.1 TalkLand

TalkLand je herná platforma, ktorá umožňuje spúšťať jednotlivé hry, nastavovať obtiažnosti výslovnosti jednotlivých hlások v hrách a vytvárať a upravovať profily detí. Platforma TalkLand je v slovenčine aj angličtine na základe hodnoty v registroch nastavenej počas inštalácie.

Do TalkLandu sa dajú hry pridávať, zatiaľ je však možné hry pridávať len počas aktualizácie aplikácie. Po spustení a ukončení hry vytvorí hra session, ktorú TalkLand nahrá na server. V prípade, neúspechu, či už kôli výpadku internetu alebo iným dôvodom sa táto session nestratí, ale ostane na disku a TalkLand sa ju pokúsi nahráť neskôr.

V rámci šprintov bola do TalkLandu dorobená funkcionálna prihlásenia používateľa, zobrazovania jeho údajov na vykonanie platby a kontroly, či používateľ už zaplatil. Do doby než používateľ zaplatí, môže TalkLand skúšať na obmedzenú dobu v skúšobnej verzii. Skúšobná verzia je prístupná aj bez prihlásenia používateľa a tým pádom bez registrácie. Ďalšou významnou pridanou funkcionálnosťou je pridávanie a upravovanie detí, ktoré bolo dovtedy dostupné iba pomocou webového rozhrania.

7.2 TalkLand media

Táto časť projektu sa začala tvoriť od druhého šprintu kedy bola spravná analýza a návrh a od tretieho šprintu sa pokračovalo v práci na jej implementácii v každom z nasledujúcich šprintov. Súčasťou TalkLand media je knižnica, ktorá poskytuje hrám v TalkLande slová alebo páry slov s jednou odlišnou hláskou v textovej forme, ich skomoleniny, obrázky opisujúce ich význam a nahrávky výslovnosti slov a ich skomolenín.

Súčasťou TalkLand media je aj aplikácia TalkLand media explorer, ktorá slúži na napĺňanie TalkLand media knižnice obsahom, čiže vytváranie, prehliadanie a mazanie balíčkov slov, slov párov slov a obrázkov. Aplikácia umožňuje tiež nahrávanie s automatickým predspracovaním nahrávok výslovnosti slov ako je odstránenie ticha z koncov nahrávky a normalizácia hlasitosti.

7.3 Goldy

Goldy je hra ktorá umožňuje v rámci platformy TalkLand cvičenie hlások s/š hravou formou. Cvičenie je priamo zaintegrovane do hernej interakcie kde hráč varuje trpaslíka pred zobudzajúcim sa drakom vyslovovaním hlásky s/š do mikrofónu.

V prípade nedostupnosti mikrofónu hra na to upozorní používateľa a vyžiada si jeho zapojenie. Hra je ošetrená aj voči neočakávaným chybám, reporty z ktorých odkladá pre TalkLand na odoslanie na server. Text v rámci hry je v angličtine/slovenčine na základe nastavenia jazyka platformy TalkLand.

V rámci šprintov bola v Goldy zjednodušená mechanika pomôcky kúzelného plášťa tak, aby viac spĺňala očakávania hráčov. Hra bola prevedená do technológie MonoGame framework, ktorá umožňuje jej prenos na ďalšie platformy. Po treťom šprinte sa v MonoGame vyskytol problém, že hra neregistrovala zvukový vstup. Tento problém bol však v nasledovnom šprinte úspešne vyriešený.

7.4 TailTrails

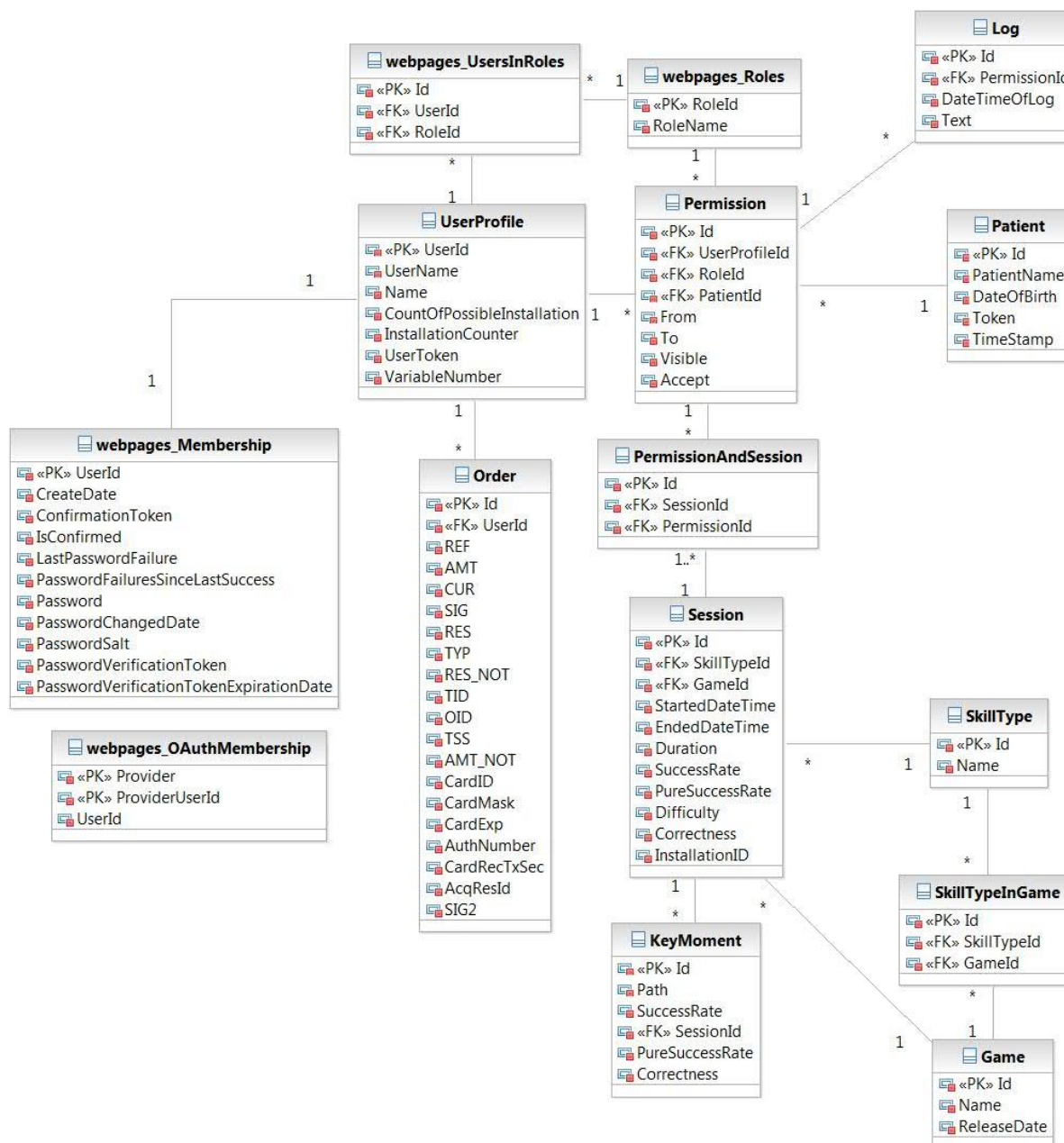
TailTrails je hra na trénovanie zvukovej diferenciácie, podobná známej hre Snake. Aktuálny prototyp umožňuje ovládanie hada a zbieranie bodov na hracej obrazovke, pričom hra sa uvedie do počiatočného stavu pokiaľ dôjde ku kolízií hada s jeho chvostom alebo okrajmi hracej plochy.

V rámci šprintov bola hra prevedená do technológie MonoGame a bol navrhnutý spôsob prístupu k informáciám slov na diferenciáciu, na ktorom je založený komponent TalkLandMedia. V piatom šprinte bola v hre doplnená možnosť výberu z dvoch hracích postáv (hadík reprezentujúci hlásku ‚s‘ a vláčík reprezentujúci hlásku ‚š‘). Do hry bolo tiež zaintegrovane využitie knižnice TalkLandMedia, pričom pri zobrazovaní bobúľ ktoré má hráč zbierať sú prehrávané nahrávky slov obsahujúcich diferencovanú hlásku, alebo hlásku k nej podobnú.

7.5 Webové rozhranie

Pomocou webovej aplikácie bežiacej na serveri môže logopéd alebo rodič sledovať priebeh cvičenia dieťaťa, ako sa dieťaťu darilo v hre a ako dlho sa hru hralo. Pozorovateľ má tiež prehľad o celkovom pokroku dieťaťa, od čoho sa môže odvíjať nasledujúci priebeh terapie.

Po založení konta rodičom si rodič v systéme vytvorí profil dieťaťa. Ak dieťa nenavštevuje logopéda, tak výsledky z cvičenia vidí len rodič, pričom rodič môže po prihlásení dieťaťa k logopédovi prepojiť hernú platformu s kontom logopéda, ktorý potvrdí prijatie nového pacienta v systéme. Od tohto okamihu vidí logopéd pacienta v systéme, a dokáže si k nemu pridávať textové záznamy z jednotlivých sedení.



Obr. 8 – Dátový model projektu Speekle

Popisy vybraných tabuliek:

Order – uchováva sa tu vytvorené objednávky používateľom. Každá objednávka je identifikovaná svojim *Id* a *REF* (unikátne evidenčné číslo objednávky). Okrem toho sa tu uchováva aj suma, mena a podpis (*AMT*, *CUR*, *SIG*). *UserId* určuje používateľa, ktorý objednávku vytvoril. Po pokuse o zrealizovanie platby príde návratový kód *RES*. Ako náhle je prijatá platba, tak po prijatí notifikácie sa dopĺňajú údaje ako id transakcie *TID*, id objednávky *OID*, či vola transakcia podpísaná *TSS*, suma, ktorá bola reálne prijatá *AMT_NOT*. V prípade, že bolo platené kartou, tak sa pridávajú aj identifikačné údaje karty *CardID*, *CardMask*, *CardExp*, *AuthNumber*, *CardRecTxSec*, *AcqResId*. Zároveň sa posiela aj podpis generovaný z týchto údajov *SIG2*.

UserProfile - slúži na správu používateľov, pričom uchováva údaje:

- *UserId* – primárny kľúč, jedinečné id používateľa

- **UserName** – reťazec zodpovedajúci e-mailovej adrese používateľa, pod ktorou sa prihlasuje do aplikácie
- **Name** – reťazec uchovávajúci reálne meno používateľa
- **CountOfPossibleInstallation** – číslo uchovávajúce počet možných inštalácií
- **InstallationCounter** – počítadlo už prebehnutých inštalácií
- **VariableNumber** – variabilné číslo unikátne pre používateľa
- **UserToken** – jedinečný reťazec 10-tich náhodne generovaných znakov pri pridávaní používateľa do systému

Patient – tabuľka slúžiaca na uchovávanie údajov o pacientoch. Každý pacient je jednoznačne určený svojím *Id* a *Tokenom*, čo je unikátny reťazec 10-tich náhodne generovaných znakov pri pridávaní pacienta do systému. Každý pacient je prepojený s logopédom alebo s rodičom pomocou tabuľky **Permission**. Okrem toho sa uchovávajú aj meno, dátum narodenia a čas vytvorenia pacienta v systéme.

Permission – spája pacienta a používateľa identifikovaného tabuľkou *UserProfile*, pričom určuje ich vzťah (*RoleId*) upresnený časovým intervalom *From* a *To*. Stĺpec *Visible* určuje, či je pacient pre používateľa viditeľný, alebo je v archíve. Stĺpec *Accept* určuje, či pacient už bol pridatý logopédom, alebo nie.

PermissionAndSession – spája tabuľky *Permission* a *Session*, pričom jeden záznam v tabuľke *Session* musí byť naviazaný minimálne na 1 riadok v tabuľke *Permission*. Každá *session* sa ukladá pre aktuálneho logopéda a pre rodičov dieťaťa.

Session – nachádzajú sa tu dáta prijaté z hernej platformy *TalkLand*. Každý záznam v tejto tabuľke predstavuje konkrétne precvičovanie dieťaťa. Je určený počiatočným (*StartedDateTime*) a koncovým (*EndedDateTime*) časom cvičenia, čistým časom (*Duration*), koľko sa dieťa skutočne hralo a je naviazaný na záznam v tabuľke *SkillType* cez *SkillTypeId* a na záznam v tabuľke *Game* cez *GameId*. Uchováva údaje o správnosti, úspešnosti a čistej úspešnosti.

7.6 API webovej aplikácie pre TalkLand

Možné URL requestov v aktuálnej verzii:

- <http://www.speekle.sk/sk-sk/SpeechTherapist/<NázovRequestu>>
- <http://staging.speekle.sk/sk-sk/SpeechTherapist/<NázovRequestu>>
- <http://beta.speekle.sk/sk-sk/SpeechTherapist/<NázovRequestu>>
- <https://www.speekle.sk/sk-sk/SpeechTherapist/<NázovRequestu>>
- <https://staging.speekle.sk/sk-sk/SpeechTherapist/<NázovRequestu>>
- <https://beta.speekle.sk/sk-sk/SpeechTherapist/<NázovRequestu>>

Jednotlivé requesty API sú podrobne popísané so všetkými možnými scenármi v nasledujúcich podkapitolách.

7.6.1 GetUserToken

Popis:

Pošlú sa správne prihlasovacie údaje používateľa, pričom mu webová aplikácia vráti jeho unikátny token a údaje potrebné na platbu.

Parametre:

- username=ondrej.galbavy@gmail.com
- password=password

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>OK</status>
  <userToken>2CD4A459-09D0-43FE-9094-6D87162E74C5</userToken>
  <userRole>SpeechTherapist</userRole>
  <variableNumber>1012546044</variableNumber>
  <account>1100606338/8410</account>
  <price>35</price>
</response>
```

Popis:

Pošle sa zlé heslo používateľa, webová aplikácia pošle odpoveď s príslušnou chybovou správou

Parametre:

- username=ondrej.galbavy@gmail.com
- password=pomylilSomSaVHesle

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>AuthenticationFailed</message>
</response>
```

Popis:

Pošle sa neexistujúce meno používateľa, webová aplikácia pošle odpoveď s príslušnou chybovou správou

Parametre:

- username=neexistujuciMail@tatoDomenaNeexistuje.com
- password=password

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>AuthenticationFailed</message>
</response>
```

Popis:

Všetko je v poriadku, ale bol prekročený povolený limit inštalácií

Parametre:

- username=ondrej.galbavy@gmail.com
- password=password

Response:

```
<?xml version="1.0" encoding="utf-16"?><response>
  <status>FAIL</status>
  <message>IncInstallationCounterFailed</message>
  <installationCount>3</installationCount>
</response>
```

Popis:

V prípade, že nastala interná chyba

Parametre:

- username=ondrej.galbavy@gmail.com
- password=password

Response:

```
<?xml version="1.0" encoding="utf-16"?><response>
  <status>FAIL</status>
</response>
```

7.6.2 HasPaid

Popis:

Pošle sa správny userToken a používateľ je evidovaný v databáze, že zaplatil

Parametre:

- userToken= 4dacc39f-b267-409b-a1f7-9de088180598

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>OK</status>
</response>
```

Popis:

Pošle sa správny userToken a používateľ je evidovaný v databáze, že nezaplatil

Parametre:

- userToken= 4dacc39f-b267-409b-a1f7-9de088180598

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
</response>
```

Popis:

Pošle sa nesprávny userToken alebo nastala interná chyba vo webovej aplikácii

Parametre:

- userToken= nespravnyUserToken

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
</response>
```

7.6.3 GetTokens

Popis:

Pošle sa správny userToken a webová aplikácia pošle zoznam childTokenov detí, ktoré prislúchajú používateľovi

Paramtere:

- userToken=4dacc39f-b267-409b-a1f7-9de088180598

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<childTokens>
  <parentUserName>mirzekdt@gmail.com</parentUserName>
  <tokens>
    <child>
      <name>Gregor</name>
      <token>5dPI9plf7z</token>
      <dateOfBirth>23.04.2008</dateOfBirth>
      <codeOfSpeechTherapist>ondrej.galbavy@gmail.com</codeOfSp
eechTherapist
    </child>
    <child>
      <name>First child of mother TalkLand</name>
      <token>ol5zwjafjZ</token>
      <dateOfBirth>23.04.2008</dateOfBirth>
      <codeOfSpeechTherapist>ondrej.galbavy@gmail.com</codeOfSp
eechTherapist>
    </child>
  </tokens>
  <result>Ok</result>
</childTokens>
```

Popis:

Pošle sa nesprávny userToken

Paramtere:

- userToken=nesprávnyUserToken

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<childTokens>
  <result>InvalidUser</result>
</childTokens>
```

Popis:

V prípade nastatia internej chyby v aplikácii

Paramtere:

- userToken=userToken

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<childTokens>
  <result>FAIL</result>
</childTokens>
```

7.6.4 DecInstallationCount

Popis:

Pošle sa správny userToken a webová aplikácia zníži počet inštalácii o 1, pokiaľ je počet inštalácii väčší ako 0.

Parametre:

- userToken= 4dacc39f-b267-409b-a1f7-9de088180598

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>OK</status>
</response>
```

Popis:

V prípade zlého userTokenu alebo internej chyby vo webovej aplikácii

Parametre:

- userToken= userToken

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
</response>
```

7.6.5 CreatePatientTalkLand

Popis:

Zašlú sa údaje o dieťati v správnom formáte, webová aplikácia vytvorí dieťa a pošle naspäť unikátny token dieťaťa.

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- userToken=4dacc39f-b267-409b-a1f7-9de088180598
- codeOfSpeechTherapist=ondrej.galbavy@gmail.com

Response:

```
<?xml version="1.0" encoding="utf-16"?>
```

```
<response>
  <status>OK</status>
  <childToken>hAvnHwHcOq</childToken>
</response>
```

Popis:

V prípade budúceho dátumu narodenia dieťaťa aplikácia posielala odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2189
- userToken=4dacc39f-b267-409b-a1f7-9de088180598
- codeOfSpeechTherapist=ondrej.galbavy@gmail.com

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>FutureDateOfBirth</message>
</response>
```

Popis:

V prípade neexistujúceho kódu logopéda aplikácia posielala odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- userToken=4dacc39f-b267-409b-a1f7-9de088180598
- codeOfSpeechTherapist=neexistujuciKodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>WrongCodeOfSpeechTherapist</message>
</response>
```

Popis:

V prípade, že sa dieťa pokúša vytvoriť logopéda, tak aplikácia posielala odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- userToken=4dacc39f-b267-409b-a1f7-9de088180598
- codeOfSpeechTherapist=KodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>RightsError</message>
</response>
```

Popis:

V prípade, že sa pošle nesprávny token používateľa, aplikácia posiela odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- userToken=nespravnyUserToken
- codeOfSpeechTherapist=KodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>WrongUserToken</message>
</response>
```

Popis:

V prípade internej chyby vo webovej aplikácii

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- userToken=userToken
- codeOfSpeechTherapist=KodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
</response>
```

7.6.6 EditPatientTalkLand

Popis:

Pošlú sa údaje o dieťati v správnom formáte

Parametre:

- childName=Ferko
- childDateOfBirth=02.02.2002
- childToken=hAvnHwHcOq
- codeOfSpeechTherapist=ondrej.galbavy@gmail.com

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>OK</status>
</response>
```

Popis:

Poše sa budúci dátum narodenia dieťaťa, aplikácia posielala odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2189
- childToken=hAvnHwHcOq
- codeOfSpeechTherapist=ondrej.galbavy@gmail.com

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>FutureDateOfBirth</message>
</response>
```

Popis:

Poše sa neexistujúci kód logopéda, aplikácia posielala odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- childToken=hAvnHwHcOq
- codeOfSpeechTherapist=neexistujuciKodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
  <message>WrongCodeOfSpeechTherapist</message>
</response>
```

Popis:

Poše sa neexistujúci token dieťaťa, aplikácia posielala odpoveď s príslušnou chybovou správou

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- childToken=neexistujuciChildToken
- codeOfSpeechTherapist=kodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
```

```
<response>
  <status>FAIL</status>
  <message>WrongChildToken</message>
</response>
```

Popis:

V prípade internej chyby vo webovej aplikácii

Parametre:

- childName=Ferko
- childDateOfBirth=01.01.2001
- childToken=childToken
- codeOfSpeechTherapist=kodLogopeda

Response:

```
<?xml version="1.0" encoding="utf-16"?>
<response>
  <status>FAIL</status>
</response>
```

7.7 Inštalátor

Inštalátor je program, ktorý inštaluje TalkLand a hru Goldy do počítača používateľa. Okrem kopírovania súborov inštalátor vytvorí odkaz na TalkLand na pracovnej ploche a záznam pre odinštaláciu. Pre správnu funkčnosť aplikácií TalkLand a Goldy inštalátor kontroluje a inštaluje aj ich závislosti: .NET 4 Framework Full Profile a XNA 4 Framework. Inštalátor notifikuje službu pre záznam udalostí o inštalovaní, aktualizácii alebo odinštalácii.

V rámci šprintov bola z inštalátora odstránená kontrola a inštalácia .NET 3.5 Framework, pretože TalkLand už nie je závislý na tejto verzii. Prebehol aj čiastočný refaktoring, odstránenie duplicit kritického kódu a zlúčenie kódu hlavného inštalátora a aktualizáčného inštalátora. Bol vykonaný prechod na kompilátor NSIS 3.0a1 s podporou Unicode.

V odinštalátore boli na konci tretieho šprintu identifikované nedostatky súvisiace s výnimočnými situáciami, napr. zlyhanie servera alebo stále spustený TalkLand.

V nasledujúcom šprinte boli vyriešené nedostatky so zlyhaním servera alebo nefunkčným sieťovým pripojením pri deaktivácii. V 5. šprinte boli vyriešené problémy s chybami pri sťahovaní závislostí alebo aktualizáčného inštalátora.

V 5. šprinte bola finálne vyriešená diakritika v inštalátore. Inštalátor TalkLand sa kompiluje s plnou podporou Unicode. Funkcia ShellExecWait, ktorá si vyžiada UAC eleváciu pri inštalácii závislostí a vracia návratový kód spusteného procesu, fungovala len v ANSI verzii. Bola nahradená upravenou verziou zásuvného modulu StdUtils pre NSIS.

7.8 Dashboard

Dashboard je webová aplikácia, ktorá bola vytvorená v rámci druhého šprintu. Poskytuje prehľad z nazbieraných dát o používaní produktov Speekle. Aplikácia poskytuje grafy pre nasledujúce typy informácií:

- Počet normálnych, neregistrovaných a anonymných sedení za jednotlivé dni

- Priebeh počtu používateľov a detí
- Počet inštalácií, aktualizácií a odinštalácií za jednotlivé dni
- Histogram – koľko používateľov vytvorilo určitý počet sedení

Webová aplikácia je vytvorená v jazyku Ruby s využitím rámca Ruby on Rails. Pripája sa priamo na databázové servery hlavnej webovej aplikácie. Aplikácia má problém s dlhodobým pripojením na databázové servery, ktoré ukončujú neaktívne spojenia. Tieto problémy vyriešené v 4. šprinte a dashboard bol nasadený na tímový server. Dashboard bol rozšírený o možnosť zobrazovanie detailných informácií o používateľoch a ich deťoch.

7.9 Služby pre TalkLand

7.9.1 Autorizačná služba TalkLand

Verejná beta verzia TalkLand pri každom spustení kontaktuje túto službu so žiadosťou o autorizovanie spustenia. Týmto spôsobom je možné verejnú beta verziu kedykoľvek ukončiť a zasielať správu o dostupnosti plnej verzie. Ďalšou významnou funkciou tejto služby je oznamovanie dostupnosti novej verzie aplikácie TalkLand.

TalkLand na server zasiela svoju verziu a náhodný reťazec, ktorý server priloží do odpovede. Odpoveď obsahuje informáciu či je dostupná aktualizácia beta verzie TalkLand a či je beta verzia stále dostupná. Odpoveď je zabalená do kontajnera, ktorý je podpísaný privátnym kľúčom. TalkLand overí pravosť podpisu odpovede verejným kľúčom. Tým je zabránené spusteniu TalkLand prehratím nahranej odpovede servera.

Autorizačná služba je automaticky kontrolovaná každých 5 minút službou uptimerobot.com.

Služba je naprogramovaná v jazyku PHP a používa databázový server MySQL. V rámci šprintov nebola služba ovplyvnená a bola presunutá na tímový server.

7.9.2 Služba pre trial tokeny

Plná verzia TalkLand umožňuje vyskúšať aplikáciu na určitý počet dní bez prihlásenia a zaplatenia. Služba generuje a overuje tokeny, ktoré majú obmedzenú platnosť. TalkLand pri prvom spustení kontaktuje službu, ktorá vygeneruje nový token v podobe náhodného reťazca. Služba si do databázy uloží dátum vygenerovania tokenu a aktuálnu dobu platnosti. TalkLand následne pri každom spustení kontaktuje službu, ktorá overí, či je daný token stále platný. Ak nie je, TalkLand bude vyžadovať prihlásenie so zaplateným účtom, inak sa nespustí.

Služba je naprogramovaná v jazyku PHP a používa databázový server MySQL. V rámci šprintov nebola služba ovplyvnená a bola presunutá na tímový server.

7.9.3 Služba pre zaznamenávanie výnimiek

Aplikácie TalkLand a Goldy vo svojom rámci implementovaný mechanizmus zachytenia neošetrených výnimiek počas behu programu. V prípade výskytu výnimky je na túto službu odoslaný chybový výpis, ktorý je následne uložený na server a zároveň je odoslaná notifikácia na určité emailové adresy.

Služba je naprogramovaná v jazyku PHP. V rámci šprintov nebola služba ovplyvnená a bola presunutá na tímový server.

7.9.4 Služba pre záznam udalostí

Inštalátor kontaktuje túto službu v prípade inštalácie, aktualizácie alebo odinštalácie. Služba zaznamenáva tieto udalosti vrátane identifikácie počítača alebo používateľa. S týmito údajmi je možné ďalej pracovať priamym prístupom do databázy. Služba je všeobecne navrhnutá pre naše potreby a môže prijímať udalosti o spustení alebo aktivácii TalkLand.

Služba je naprogramovaná v jazyku PHP a používa databázový server MySQL. Služba bola vytvorená popri refaktoringu inštalátora a bola presunutá na tímový server.