

TDK DOLGOZAT

Csányi Ottó

2013

Budapesti Műszaki és Gazdaságtudományi Egyetem

Villamosmérnöki és Informatikai Kar

Vállalat információs rendszerek szakirány



„Gemifikáció” a vállalati szoftverek területén: Alkalmazás töltési ideje alatt megjelenő tartalom

Konzulens:

Dr. Szikora Béla
egyetemi docens

Készítette:

Csányi Ottó

2013

Tartalomjegyzék

1. Célkitűzés.....	4
2. Bevezetés.....	5
3. Mi a „gemifikáció”?	6
3.1 Történelem	6
3.2 Összegzés.....	8
4. „Gemifikáció” az SAP-nál.....	9
4.1 Sikeres próbálkozások	10
4.2 Összegzés.....	12
5. „Gemifikáció” töltésidő alatt: tények, vélemények, saját ötletek.....	14
6. Az első prototípus létrehozása	18
6.1 Környezet: SAP HR Approvals Android alkalmazás	18
6.2 Megvalósítandó ötlet	20
6.3 RSS feed technológia	21
6.4.1 Mock up.....	22
6.4.2 Use-case	23
6.4.3 Modell kidolgozás.....	23
6.4.3.1 Osztályok kidolgozása.....	23
6.4.3.2 Osztálydiagramok	26
6.4.3.3 Szekvencia diagramok	27
6.5 Szkeletonból prototípus	28
7. Felmérés és eredmények tervezése	30
7.1 Felmérés	30
7.2 Eredmények.....	30
Irodalomjegyzék	32
Terminológia.....	33
Mellékletek.....	35

1. Célkitűzés

A feladat célja kutatás a „gemifikáció” területén. A kutató munkához szükség van létező „gemifikációs” megoldások megismerésére illetve egy saját prototípus elkészítésére. A konkrét prototípus egy olyan mini alkalmazás, amely egy program eddig kihasználatlan töltésideje alatt „gemifikációval” tartja fenn a felhasználó figyelmét. Az alkalmazáshoz szükség van egy keretrendszerre, amely menedzseli a tartalmat és továbbítja azt a töltésidőben felugró ablak számára. A prototípus tesztalanyok segítségével visszacsatolást képes nyújtani a következő fontos szempontokról:

- **Figyelem fenntartásának mértéke különféle „gemifikációs” tartalom esetén és nélküle.**
- **Töltésidő hosszának érzékelése miközben unatkozik a felhasználó, illetve ha foglalkoztatva van.**
- **Hajlandóság a töltésidő hosszának növelésre, amennyiben a felhasználó számára hasznos adatot jelenítünk meg.**
- **Az egyes tartalmak összevetése (pl. interaktív vagy passzív) előnyeinek és hátrányainak összevetése.**
- **Van-e egyáltalán helye „gemifikációnak” üzleti alkalmazás töltésideje alatt?**

A cél egy olyan prototípus megalkotása, amellyel oda lehet menni végfelhasználókhoz és a prototípus kipróbálása után fel lehet tenni a fenti kérdéseket. A kérdések feltétele illetve az azokra kapott válaszok elemzése, eredményének közzétevése terjedelmi és idő korlátok miatt nem tartalma a dolgozatnak, azonban egy nagyobb terjedelmű munkában (szakdolgozat) erre is sor kerül.

2. Bevezetés

A „gemifikáció” nem más, mint játékos felfogás illetve játékos elemek beültetése egy olyan szoftverbe, amelyet eredendően nem azért hoztak létre, hogy játszanak vele. Ezzel elősegítheti a felhasználó elkötelezettségét, továbbá serkenti a problémamegoldó képességet, nem utolsósorban a felhasználó megfélekedezik az alkalmazással töltött időről.

A mai vállalati alkalmazások a megrendelői igények lehető leggyorsabb kielégítésére fókuszálnak, így a „gemifikáció” beiktatása legtöbb esetben alacsony prioritást élvez és kimarad a végtermékből.

Dolgozatom ismerteti a „gemifikáció” alapelveit, az ezt kihasználó lehetséges alkalmazásokat a vállalatirányítási rendszerek körében. A kutató munka még a fejlesztési lépések előtt foglalkozik azzal, hogy van-e értelme „gemifikáció” használatának ilyen környezetben. Ezt a vizsgálatot a korábbi alkalmazások megismerésén kívül a vállalati információs alkalmazás fejlesztőivel és felhasználóival készített interjúk, fórumban feltett kérdések és azokra válaszok segítik.

Egy programot részletesen is bemutatok, amelynek célja: „Vállalati alkalmazás betöltési idejének kihasználása”. A program elkészítésének indoka, hogy a felhasználó töltési (vagy Windows-os környezetben homokóra) időben unatkozik. A cél, hogy ne unatkozzon, a figyelme le legyen kötve, ezáltal gyorsabban teljen el a várakozási idő.

Ez egy jól bevett szokás a PC-s és egyéb platformra készülő játékok körében, amikor a következő pályát vagy korábbi mentést kell betölteni. Ilyenkor a következő pályáról tudhatunk meg hasznos információkat vagy a játékról egyéb tanácsokat, amelyek a kezdő játékosok számára ismeretlenek lehetnek és tanulásukat szolgálják.

Hasonló tanácsokat, tanító jellegű tippeket, figyelemfelkeltő, interaktív tartalmat megjeleníthetünk egy üzleti alkalmazás felhasználói számára is várakozási időben. Ezt azonban még az egyik legnagyobb vállalati információs rendszereket gyártó cég,; az SAP nem alkalmazta. Egy magyar fejlesztésű SAP mobil alkalmazás (HR Approvalas) lehetőséget nyújt arra, hogy az általam fejlesztett prototípust integrálva valós üzleti alkalmazást adjunk olyan tesztalanyok kezébe, akik véleménye segíthet eldönteni a témával kapcsolatos nyitott kérdéseket.

3. Mi a „gemifikáció”?

A „gemifikáció” nem más, mint játékos felfogás beültetése egy olyan szoftverbe, amelyet nem azért használnak, hogy játszanak vele. Ezzel elősegítheti a felhasználó elkötelezettségét, továbbá serkenti a problémamegoldó képességet, illetve a felhasználó megelégedik az alkalmazással töltött időről a „gemifikációnak” köszönhetően.

Az emberekben erős versenyszellem él, ezt illusztrálja az 1. ábra. A „gemifikációs” technikákat tipikusan úgy alkalmazzák, hogy serkentse ezt a versenyszellemet, a kihívásokban való részvételt, a teljesítmény elérését, a játékos értelemben vett státusz elérését és az önzetlenséget. Az olyan típusú megoldások, mint a felhasználó pontokkal való jutalmazása után kapott virtuális díjazás, kitüntetés vagy szintnövelés szintén a „gemifikációs” stratégiák részei.

A másokkal szembenő versengés szintén fontos eleme a számítógépes játékoknak, amit a „gemifikáció” területén is alkalmazhatunk. Egyes felhasználók által elnyert díjak láthatóvá tétele más felhasználók számára vagy a legügyesebb felhasználók publikus ranglistán való feltüntetése más felhasználókat is a versenyben való részvételre ösztökélhet.

Egy további előfordulási módja a „gemifikációnak” a már létező feladatok, funkciók játékosabbá tétele. Ezt elősegítő technikák például „tutoriólok” beiktatása, kihívás növelése, narrátor hozzáadása, amely végig kíséri a felhasználót az adott folyamatokon keresztül.[1]



1. kép: Ösztönös gyermeki arckifejezés számítógépes játék közben

3.1 Történelem

Bár a „gemifikáció” (angolul gamification) kifejezést először 2002-ben Nick Pelling használta, mégsem vált népszerűvé 2010-ig. Még mielőtt ez a kifejezés értelmet nyert volna, már más

területeken is felbukkantak a „gemifikáció” elemei, például a tudományos célra felhasznált vizualizációk adaptálták a videó játékok egyes elemeit. A „Gemifikáció Nagypjának” Charles Coonradt-t szokás emlegetni, aki 1973-ban megalapította a „The Game of Work” nevezetű tanácsot és 1984-ben könyvet írt [2], melynek ugyan ezt a nevet adta.

A „gemifikáció” mint kifejezés 2010-ben terjedt el igazán, amikor már sokkal specifikáltabb jelentéssel bírt. Ez a terület felkeltette a kockázatvállaló befektetők érdeklődését, volt aki a játékok legnagyobb sikerekkel kecsegtető jövőbeli területének vélte.

Számos kutató úgy gondolja, hogy a „gemifikáció” eredete olyan korábbi kísérletekre vezethető vissza melyek megpróbálták adaptálni játékos elemeket olyan környezetbe, melyet nem a játékoság jellemzett. Egy ilyen amerikai menedzsment trend a „fun at work”.

A technológiai megoldásokkal foglalkozó cégek esetén számos üzlet során keletkeztek „gemifikációs” platformok. 2007 októberében a Bunchball nevezetű cég –melyet az Adobe Systems Inc. támogatott- volt az első vállalat, amely játékos megoldásokat szolgáltatásként ajánlott az Office című sorozat közösségi oldala felé. 2009 júniusában alapult meg a BigDoor nevezetű „start up” cég, mely „gemifikációs” technológiákat nyújtott nem játékokkal foglalkozó weboldalak számára. 2010-ben létre jött a „Badgevill” nevezetű cég, amely szintén számos megoldással állt elő, hogy növeljék az ügyfelek hűségét és a dolgozók elhivatottságát. Később olyan nagy vállalatok is elkezdték a „gemifikáció” beiktatását, mint a SAP AG, IBM, EMC, Google (lásd 2. kép), Deloitte stb...



2. kép: A 15 éves szülinapot ünneplő Google „gemifikációja”

Akadnak kritikus vélemények is a „gemifikációval” szemben. Sebastian Deterding a University of Hamburg kutatója úgy vélekedik a jelenlegi „gemifikációs” stratégiákról, hogy azok túlságosan erőltetett, mesterséges módon igyekeznek a teljesítményt növelni és nem élvezetesekek. A Massachusetts Institute of Technology professzora Kevin Slavin úgy gondolja, hogy „gemifikáció” területén folytatott üzleti kutatások eredményeként létrejött alkalmazások félrevezetőek azon felhasználók számára akik nem mozognak otthonosan a számítógépes játékok világában.[1]

3.2 Összegzés

Az alapelvek megismerése után a következő szempontokat érdemes figyelembe venni egy saját alkalmazás vagy bővítmény fejlesztése során:

- Törekednünk kell arra, hogy a „gemifikációnk” hozzájáruljon ahhoz, hogy az üzleti alkalmazással eltöltött időt a felhasználó könnyen elillanónak érezze.
- Ezt a hatást igyekezni kell olyan elemek beiktatásával elérni, amelyek fenntartják a figyelmet és előhózzák az emberekben lakozó versenyszellemet.
- A versenyszellemet a felhasználó jutalmazásával és az eredmények publikussá tételével fokozhatjuk
- A kritikus véleményekből is építkezni kell:
 - vigyázni kell, nehogy egy olyan tartalommal, játékkal álljunk elő, amely egyáltalán nem is élvezetes rontva ezzel az üzleti alkalmazás színvonalát,
 - nem kell minden áron „játszható” megoldásra gondolni, hiszen nem feltétlen minden üzleti alkalmazás felhasználója élvezi a számítógépes játékokat,
 - egyes vélemények szerint semmilyen „gemifikációnak” nincs helye üzleti alkalmazásban, velük is számolni kell.

4. „Gemifikáció” az SAP-nál

A legfontosabb kérdés: Vajon van-e helye „gemifikációnak” az üzleti alkalmazások világában?

A kérdés megválaszolásához elengedhetetlen az emberek véleményének meghallgatása illetve létező hasonló programok sikerességének kiértékelése.

Az SAP dolgozóinak (fejlesztők és termékek technikai támogatói) és egyéb a vállalat információs rendszerek világában jártas személyeinek véleménye megoszlik. Van olyan, aki remek gondolatnak találja a „gemifikáció” használatát az üzleti alkalmazásban és persze vannak kételkedők. Felmérésem szerint a leggyakoribb érvek, ellenérvek a következők:

Pozitív hozzáállás: Általában aki jó ötletnek tartja, egyből el kezd gondolkodni saját ötletein vagy az én ötleteimhez tesz javaslatot. Látszik, hogy érdeklő, érdeklődve fogadna egy ilyen megoldást. Ezekre például az SAP-nál dolgozó menedzserrel **[1. melléklet]** és technikai terméktámogatóval **[2. melléklet]** készített interjúval mutatok.

Negatív hozzáállás: Szerintük nincs helye üzleti alkalmazásban „gemifikációnak”. Fejlesztési szemszögből felesleges energia kiadásnak találják, minőségi szempontból szerintük az üzleti alkalmazás robusztusságának rovására mehet a „gemifikáció”. Kompetens személyek véleménye szerint **[3. melléklet]** a „heavy userok” esetében ezen megoldások egy idő után a felhasználókat az örületbe kergetnék. Ugyanakkor szerintük is van néhány megfontolható eset. Ilyen például a hosszabb ideig „unatkozó” felhasználó az anyagszükséglet tervezés (MRP – Material Requirements Planning) futtatása alatt.

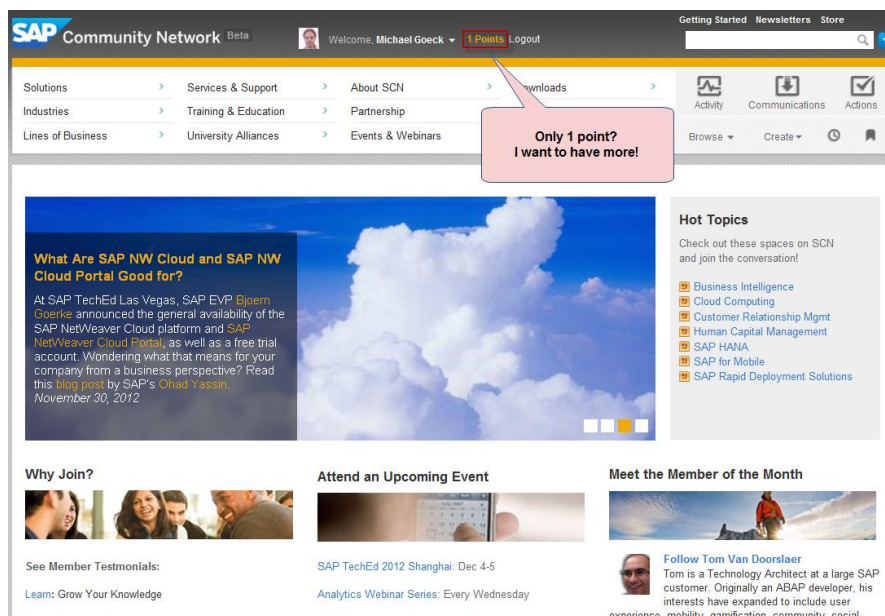
Tehát ezekből a kritikus véleményekből kiindulva, egy ilyen programnak könnyedén integrálhatónak és eltávolíthatónak kell lennie. Innovatív, letisztult, könnyedén leprogramozható megoldásokra van szükség azért, hogy a fejlesztők idejét ne nagyon rabolja el. Az erőforrások minimális idejű foglalása szintén fontos szempont. A fentiek alapján a „gemifikáció” jobbára olyan alkalmazásokban lehet releváns, ahol „consumer” típusú felhasználóink vannak.

4.1 Sikeres próbálkozások

Bár ez egy új irányzat, a SAP-nál már közel 100 „gemifikációs” kezdeményezés létezik, ezek közül néhányat alább részletezek, melyek a legsikeresebbek vagy a saját ötletemhez is jó adalékként szolgálnak [3]:

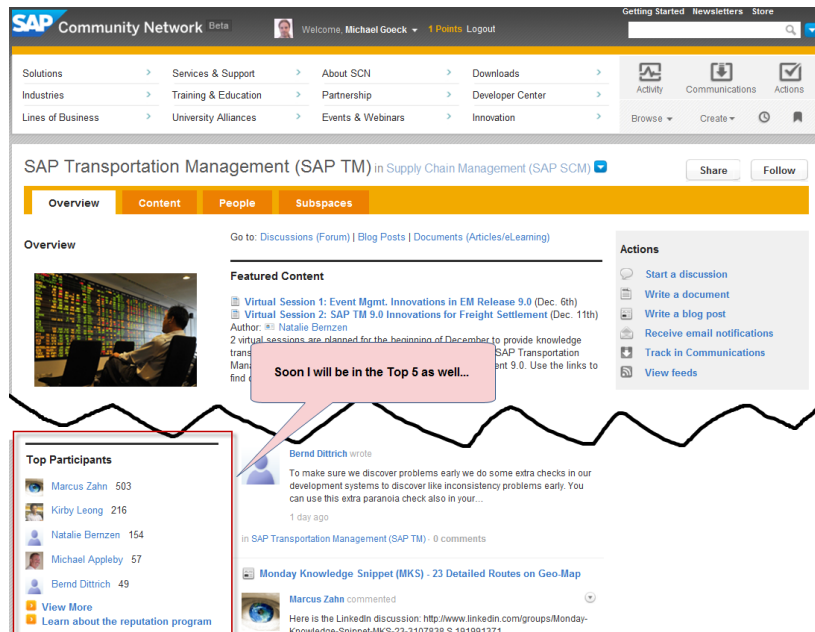
SCN (SAP Community Network)

Az SAP dolgozói; fejlesztői, üzleti folyamatok szakértői pontokat szerezhetnek, ha hasznos információkkal bővítik a Community Network-öt. Ilyen pontszerzési lehetőség lehet például egy hasznos fórum bejegyzés. A pontok arányában a felhasználókat státuszokkal, medálokkal jutalmazzák. Az oldalt és a pontgyűjtést a 3. ábra szemlélteti.



3. kép: A felhasználók pontokat szerezhetnek

Minden közösség legtöbb ponttal rendelkező tagjai egy publikus ranglistán is szerepelnek, arra ösztökélve ezzel a többi felhasználót, hogy minél több pont begyűjtésével felkerüljenek a ranglistára. Egy ranglistát a 4. kép mutat be.



4. kép: Ranglista

Movies SQL Game

HANA alapú SQL tanulást elősegítő játék. A szabályok egyszerűek, ha csatlakozik a játékos, akkor összekerül egy ellenféllel. Az alkalmazás filmekkel kapcsolatos kérdéseket tesz fel és a játékosoknak be kell írniuk azt az SQL lekérdezést, amely megadja a választ. Amennyiben jó a megoldás akkor az aktuális kör megnyerése lesz a helyes megfejtő jutalma. Aki először nyer 2 kört, azaz előbb add meg két korrekt lekérdezést, mint ellenfele az nyert. A lekérdezés megalkotására 3 perc áll rendelkezésre.

EcoDriving Mobile app

Havi rendszerességgel küldött hír az autótulajdonosok számára, amelyből a céges autójukkal megtehető további kilométerekről tájékozódhatnak.

HANA In-Memory for iPhone

A cél, hogy tájékoztassuk a kollégákat és ügyfeleket az új HANA / In Memory technológia kulcsfontosságú újításairól. Ez az alkalmazás elmagyarázza a technológia előnyeit játékos módon.

Wellness @ work

Egyszerű „gemifikációs” projekt, amely arra ösztönzi a cég dolgozóit, hogy egészséges életmódot folytassanak. Eredményként díjakat, jutalmakat nyerhetnek. Az eredményeket az alkalmazottak egymás közt összevethetik, így a program felhasználói közt egészséges versenyszellemet alakít ki. Az 5. ábra szemlélteti az alkalmazás felhasználói felületét.



5. kép: Wellness@Work alkalmazás

SAP álláskereső portál:

Az SAP-nál világszerte meghirdetett állások vannak egy-egy ponttal jelölve egy térképen. A térkép Adobe Flash alapú és jó néhány látványos funkcionalitással bír. Például egyetlen kattintással kijelölhetjük a térképen az egyes országokat, amelyekben található nyitott pozíciók csoportosítva listázódnak ki. A térkép látványosságával és sokrétű funkcionalitásával felelteti a felhasználóval a portálon töltött időt, amit saját magamon tapasztaltam állásbongészés közben.

Position Title	Location	Work Area
Senior Product Sales Executive (Mobility)	GA, Atlanta / United States of America	Sales
Senior Product Sales Executive (Data to Decision)	GA, Atlanta / United States of America	Sales
Technical Expert in Center of Expertise	Budapest / Hungary	Customer Service and Support

6. kép: SAP álláskereső portál; kijelölt országok szemléltetése

4.2 Összegzés

A fenti alkalmazások megismerése után megfigyelhető, hogy két főbb technológiai irányzata van a „gemifikációnak”, az egyik a szimplán felhasználói interfész ötletes látványával váltja ki az alkalmazáshoz való kötődést. Ilyen például a bemutatott SAP álláskereső portálon használt térképes megoldás. A másik iránynál pedig már szervertől megjelenik a gemifikáció, ilyen például a felhasználókhöz rendelt SCN Community pontok és rangok.

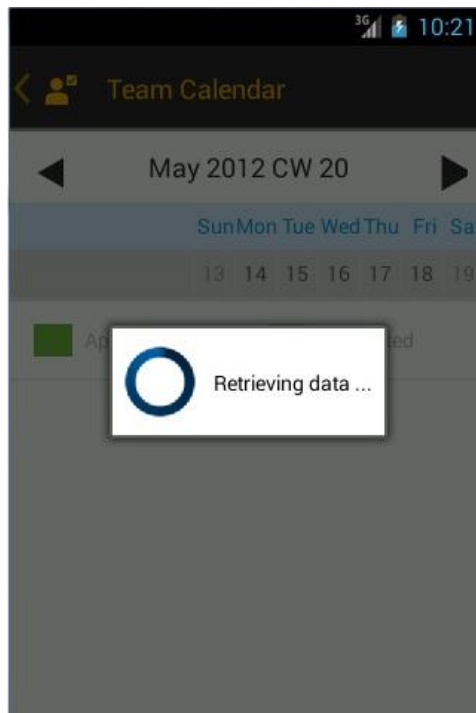
Ezekből az ötletekből a két legegyszerűbb akár töltésidő tartalmaként is elképzelhető a saját fejlesztésem keretein belül. Az autó üzemanyag keretéről való tájékoztatást továbbgondolva, mi lenne, ha a felhasználó számára releváns rövid információkat jelenítenék meg. Az ilyen

jellegű „hírekre” RSS segítségével HTTP kapcsolaton keresztül feliratkozhatunk. A másik megfontolandó ötlet a HANA SQL kvízből ered, érdekes irányelv annak a vizsgálat, hogyan reagálnának a felhasználók interaktív kvízekre.

5. „Gemifikáció” töltésidő alatt: tények, vélemények, saját ötletek

Üzleti alkalmazás töltési ideje alatt értjük azt az eltelt időt, amíg az aktuális üzleti funkciókkal, jelentéssel bíró képernyő után betöltődik a következő üzleti funkcionalitásokat ábrázoló képernyő. Egyes vélemények szerint (lásd fórum bejegyzések [4]), ennek az időnek néhány tized másodpercben mérhetőnek kellene lennie és e felé kell törekedni az összes.

Ezzel ellentmond a SAP legújabb fejlesztési keretrendszerének (Fiori) problémás betöltési ideje [5]. A SAP Fiori alkalmazások egy letisztult, intuitív felhasználói felülettel rendelkeznek, a böngészőben futó JavaScript kód építi fel az alkalmazást. [6] A korszerű technológiát használó alkalmazások azonban van egy kezdeti cash memóriába töltési ideje, ami 3G-s internet esetén 35 másodperc.[5] Tehát, még egy modern keretrendszerre épülő alkalmazás esetén is van értelme elgondolkozni azon, hogyan helyettesítenénk a töltési időt jelző „homokórát”.



7. kép, alkalmazás képernyő töltési idő alatt

Számolnunk kell azonban a következő várakozási idővel kapcsolatos pszichológiai tényekkel is, amelyek a töltési idő alatt szintén relevánsak:[7]

- Az az idő, amikor valaki unatkozik, nincs foglalkoztatva, hosszabbnak tűnik, mint ha foglalkoztatva telik ugyan az az idő.
- Bizonytalan ideig tartó várakozási idő hosszabbnak tűnik, mint az előre közölt várakozási idő.
- Nem megmagyarázott várakozási idők hosszabbnak tűnnek, mint az olyan várakozások, amely létrejöttének oka tisztázott.
- Az aggodalom, szorongás növeli a várakozási idő időtartamának érzetét.

- Minél értékesebb egy szolgáltatás annál többet hajlandó várni egy ügyfél.

Mindennapi munkám során a SAP Supplier Relationship Management (SRM) programjával foglalkozok. Bár ez egy alapvetően desktop alkalmazás (már létezik mobilon futó verziója is), mégis a nagy adatmennyiségek betöltése miatt sokszor 10 másodperc fölötti a töltésidő. Ez az idő arra pont kevés, hogy megnyissak egy böngészőt és „manuálisan gemifikáljam” saját magamat, de arra sok, hogy ezt az időt a homokóra figyelésével töltssem. Innen jött az ötlet, mi lenne, ha én, mint kezdő felhasználó a homokóra helyett a meglehetősen összetett beszerzési program működéseinek részleteit, hasznos tippeket látnék.

A „gemifikáció” alapelvei és néhány ilyen létező alkalmazás megvizsgálása után, saját kivitelezhető ötleteken kezdtem el gondolkodni. Kezdetben volt a töltésidőn kívül más „gemifikációs” ötletem is. A lényeg, hogy az általam fejlesztett program, az SAP valamely üzleti alkalmazásán belül implementálható legyen. Nagyon tetszett például az előző fejezetben ismertetett SAP állskereső honlapján található Adobe Flash térkép. Csak azt szerettem volna megnézni, hogy hány nyitott pozíció van jelenleg a magyarországi lokációnál, de közel fél órát ott ragadtam a honlapon és élvezettel próbálgattam a térkép funkcióit. Úgy gondoltam egy üzleti alkalmazáson belül is jó lenne használni egy ilyen térképet, amely például szállítási címek pontjait jelenítené meg és az egyes csomópontokra kattintva bővebb szállítási információt mutatna meg. Egy másik ötletem az volt, hogy az üzleti alkalmazásoknál gyakori töltés időt használjuk ki és a homokóra helyett valamilyen hasznos adatot lásson a felhasználó, ne unatkozzon. Az utóbbi ötletem mellett döntöttem végül és annak a megvalósítását kezdtem el kidolgozni.

Ezen döntés mellett, többek között az szólt, hogy bármely üzleti alkalmazásnál releváns lehet, de elég csak egynél rámutatni, hogy a „gemifikáció” sikeresen alkalmazható töltési idő alatt. Szintén az ötlet mellett szólt, hogy számtalan tartalom elképzelhető az egyszerűtől az összetettebbig, ezek közül néhány ötletet alább ismertetek:

Hasznos tanácsok, tippek elhelyezése töltésidőben megjelenő ablakban.

- Pozitív gondolatok az ötlettel kapcsolatosan:
 - Taníthatjuk a program felhasználóit.
 - Olyan funkciókról, trükkökről olvashatnak, melyek a program komplexitásából adódóan eddig lehet, hogy ismeretlen volt számukra.
 - Rövid töltési idő esetén is lehet értelme.
- Negatív gondolatok az ötlettel kapcsolatosan:
 - Amennyiben alkalmazásfüggő információkban gondolkozunk, akkor az egyszerűbb alkalmazásoknál felesleges lehet ez a tartalom.

Kérdezzük le az alkalmazáson keresztül a GPS koordinátákat, majd ezek segítségével jelenítsünk meg az aktuális helyről információt.

- Pozitív gondolatok az ötlettel kapcsolatosan:

- Itt is tanulhat a felhasználó, igaz nem az alkalmazásról, de ez lehet számára „szórakoztatóbb” információ, ami jobban fenntartja a figyelmet a szoftver használata során.
- A tartalom széles palettája elképzelhető, például:
 - képek az aktuális településről,
 - nevezetességek, érdekességek,
 - időjárási adatok.
- Rövid töltési idő esetén is lehet értelme.
- Negatív gondolatok az ötlettel kapcsolatosan:
 - Van-e értelme olyan adatok megjelenítésének (például időjárás), amit egyébként is pár érintéssel megnézhet a felhasználó az adott mobil készüléken.

Kvíz játék.

- Pozitív gondolatok az ötlettel kapcsolatosan:
 - Interaktív.
 - Szórakoztatóan telik a várakozási idő.
 - Lehet alkalmazásfüggő és alkalmazás független kvíz is.
 - A felhasználó pontokat is szerezhet, a pontok alapján pedig akár virtuális rangot, ennek publikussá tétele pedig a többi felhasználót is a pontgyűjtésre ösztökélheti.
- Negatív gondolatok az ötlettel kapcsolatosan:
 - Van-e erre elég idő?
 - Erőforrás?

Igaz-hamis

Hasonló előnyök és hátrányok mint a kvíz esetén.

Nyelvtanulás

- Pozitív gondolatok az ötlettel kapcsolatosan:
 - A felhasználó kiválaszthatja milyen nyelven szeretne tanulni töltési idő alatt.
 - Adott egy címke amelyre a felhasználó anyanyelvén fel van írva egy szó és adott egy szövegdoboz amiben várjuk a szó jelentését az adott idegen nyelven, ha talált, akkor ezt már nem vagy ritkábban látja viszont a felhasználó. A helyes megfejtés minden esetben megjelenítésre kerül.
 - Szintezhetőség: kezdő, haladó, profi.
- Negatív gondolatok az ötlettel kapcsolatosan:
 - Van-e erre elég idő?
 - Erőforrás?

- Jó ötlet-e nyelvet tanulni üzleti alkalmazás használata közben?

Diagram az alkalmazottak hátralévő szabadnapjainak számáról.

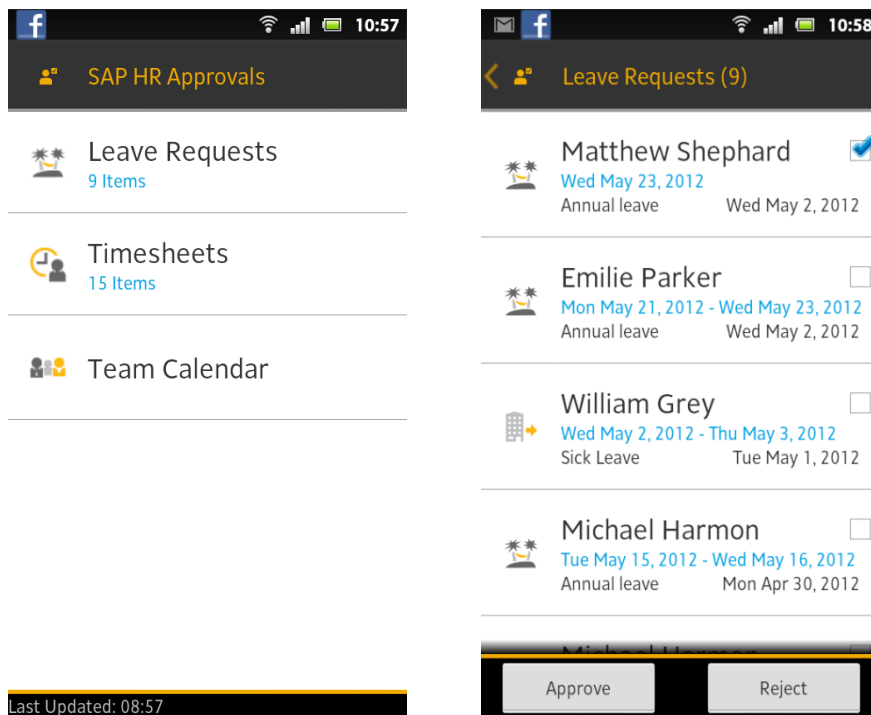
- Pozitív gondolatok az ötlettel kapcsolatosan:
Az üzleti alkalmazást használó menedzsereknek tetszene.
- Negatív gondolatok az ötlettel kapcsolatosan:
Túl nagy teljesítmény igény.

6. Az első prototípus létrehozása

A cél, hogy ebben a részben rámutassak arra, hogyan kell egy Rss feed XML válaszát egy Android dialógus ablakban megjeleníteni. Ez alapján bármely Android alkalmazás várakozási idejében felbukkanó dialógus ablakát felruházhatjuk ezzel a tartalommal. A prototípus a SAP corporate portál híroldalának címei-t jeleníti meg. A valós üzleti alkalmazás, amellyel lehetőségem van összekapcsolni a saját mini töltésidő alkalmazásomat a SAP HR Approvals nevű Android alkalmazása. A fejezetet az alkalmazás ismertetésével kezdem.

6.1 Környezet: SAP HR Approvals Android alkalmazás

Az alkalmazás, amin belül integrálhatom és tesztelhetem a saját töltésidő dialógusomat az SAP HR Approvals programja. Maga az alkalmazás csoportokat irányító menedzserek számára készült.

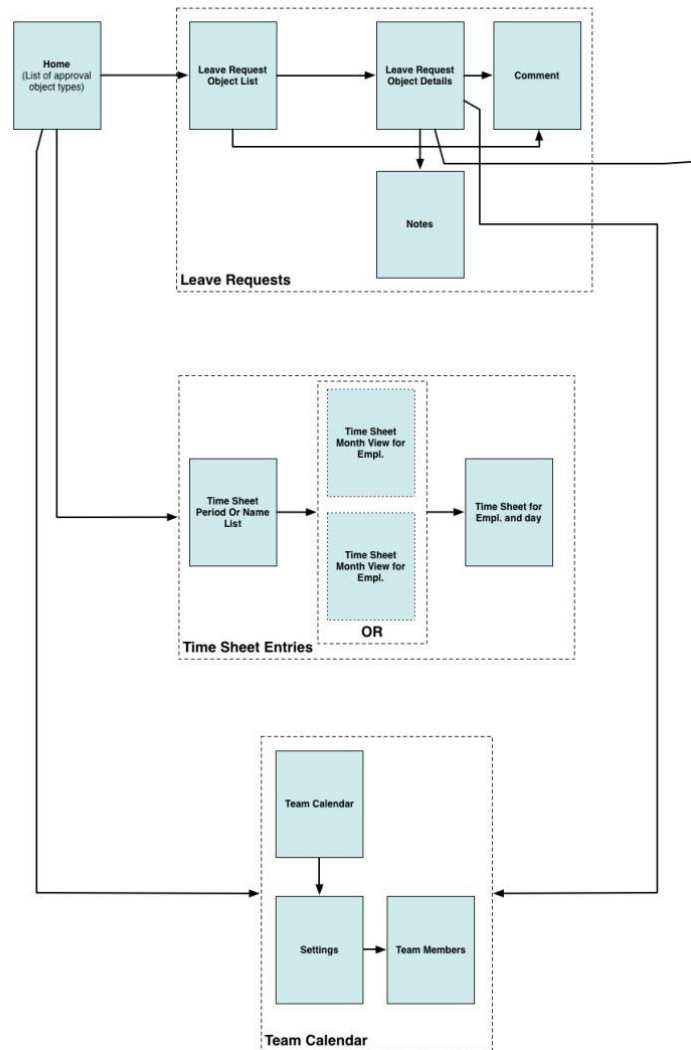


8. kép: SAP HR Approvals

Az alkalmazásnak 3 fő funkciója van, amelyekre a kezdő képernyőről navigálhatunk el:

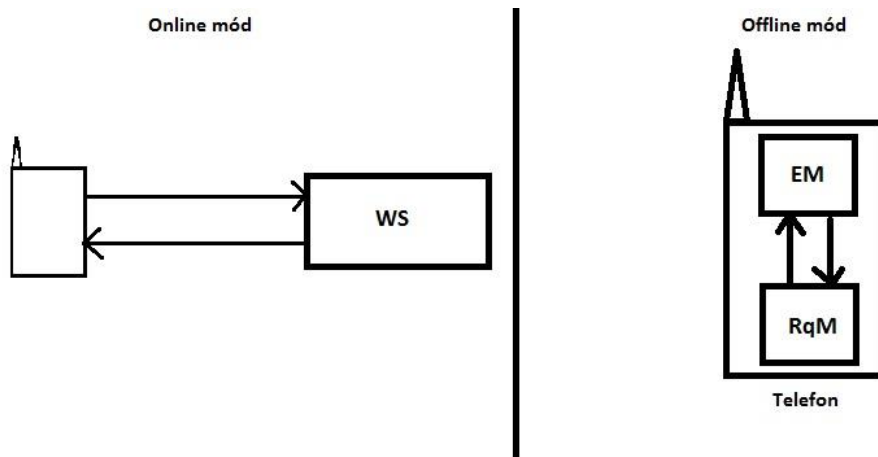
- Szabadságok jóváhagyása (Leave Requests): Miután ide navigáltunk először a szabadságolási kérvények listáját látjuk. Az egyes kérvényeket a menedzserek jóváhagyhatják vagy elutasíthatják, ekkor megjegyzést fűzhetnek a döntésükhöz vagy feljegyzést a kérvényről.
- Munkabeosztások ellenőrzése (Timesheets): Szintén a kezdő képernyőről választható ki ezt a funkciót is. Egy listában megjelennek az alkalmazottak nevei, valamelyik nevet kiválasztva az alkalmazott napi szinten lebontott tevékenységét tudják ellenőrizni a menedzserek.

- Csapat naptár (Team Calendar): Itt egy képernyőn heti nézetben tekinthető meg a csapat egyes tagjainak tevékenysége.



9. kép: A program nézetei és funkciói közt fennálló szekvenciák

A program rendelkezik online és offline móddal. Ahhoz, hogy eladásakor egy demóalkalmazást be tudjanak mutatni, elegendő egy backend és webszerver elérés nélkül példaadatokkal feltöltött offline módban üzemelő alkalmazás. Ebben a módban az EntityManager és a RequestManager között az alkalmazáson belül történik a kommunikáció, tehát nincs online kérésindítás valamely webszerver felé. Miután egy valós cég megvásárolja az alkalmazást, nekik be kell állítaniuk a backendet amit valós felhasználók a programba való bejelentkezés után elérnek, így menedzselhetik a saját csapataiktól érkező szabadság igényléseket.



10. kép: Online és offline mód

Számomra az egyszerű módosíthatóság és tesztelés offline módban ideális, hiszen ekkor nem kell rendelkezni felhasználóval és valós backend rendszerrel, akár a saját telefonomon is futtathatom az alkalmazást ebben a módban. Itt lehetőség van a töltésidő manuális meghatározásához is, ami az általam kitűzött felmérések (lásd 7.1 alfejezet) elkészítéséhez elegendő. Azonban ahhoz hogy a tényleges várakozási időket jobban megvizsgáljuk, esetleg megjósoljuk, mikor vannak a tipikusan hosszabb várakozási idők az offline mód nem elegendő.

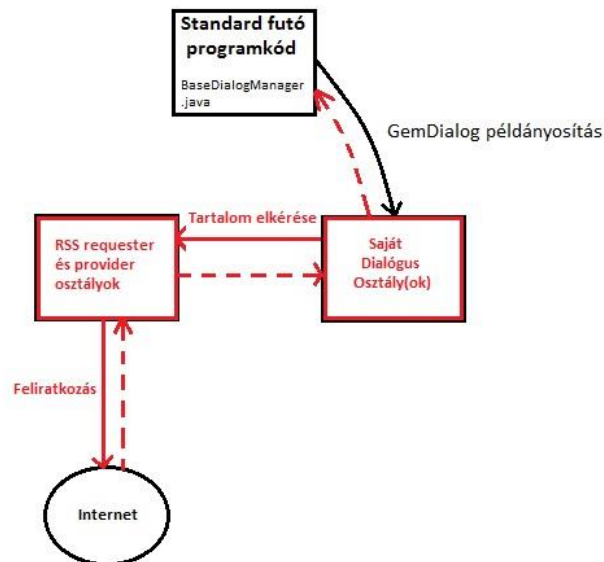
6.2 Megvalósítandó ötlet

Az alapötlet, ami a töltésidő alatt megjelenő „gemifikáció”. Ehhez próbáltam a lehetőségek megvizsgálása után olyan tartalmat választani, amely szerintem hosszú - és rövidtávon is eléri a célját, azaz hogy; fenntartsa a felhasználó figyelmét töltési idő alatt. Az interaktív megoldás a megvalósítás részletesebb vizsgálata nélkül is belátható, hogy plusz erőforrásokat köt le, ezáltal amíg tart az interakció a felhasználó és a UI közt ugyan úgy feltartjuk a felhasználót. Éppen ezért csak akkor lehet hasznos egy interaktív megoldás, ha teljesen leköti a felhasználó figyelmét és hosszabb távon nem válik unalmassá illetve idegesítővé. Ezzel szemben a hírek megjelenítése első megközelítésben hosszabb távon is megállhatja a helyét, hiszen folyamatosan frissülő adathalmazról van szó. Ezt bebizonyítani azonban csak több tesztalany véleményére támaszkodva lehet, amihez szükség van egy kézzel fogható prototípusra.

A prototípusnak az alábbi technológiai követelményeknek kell megfelelnie:

- Példányosításával integrálható legyen bármely Android alkalmazáson belül (az operációs rendszer verziószámaival való vizsgálatától eltekintve.)
- RSS technológiát alkalmazva töltésidőben jelenítsen meg híreket.
- Rugalmasan alakítható legyen, ha az aktuálison kívül más RSS feed XML tartalmát kell feldolgozni.
- Legyen lehetőség az RSS feed-ek kibővíthetőségére.
- Legyen lehetőség más fejlesztők töltésidőben elképzelt tartalmának fogadására.
- Az RSS feed vagy feed-ek illetve kívülről kapott tartalmak közül menedzselje, hogy mit továbbít a felhasználó felé.

Első megközelítésben tehát adott egy futó üzleti Android alkalmazás, ami éppen a következő képernyőt szeretné betölteni, ezért létrehoz egy várakozást jelző dialógust, hogy az töltési idő alatt megjelenjen. Ez a dialógus azonban nem a homokóra animációt választja tartalmául, hanem ennél hasznosabb tartalmat; a vállalattal kapcsolatos híreket. (lásd az x. vázlatos ábrát).



11. kép: Kezdetleges elképzelés a kommunikációról

Az Android dialógus és az ebben elhelyezett Rss feed segítségével kapott tartalom együttes előnyei:

- A dialógus előnye önmagában, hogy figyelemfelkeltő hatása van, hiszen hirtelen előbukkanó UI elemről van szó. Ráadásul a háttérben lévő elemek elhalványulnak, így a felhasználó csak a dialógusban lévő tartalomra fókuszál.
- Ezt a tartalmat folyamatosan frissülő hírek töltik ki, így elkerülhető, hogy ugyan azt a tartalmat gyakran ismétlődve lássa a felhasználó.
- Minden nagyobb cégnek vannak céggel kapcsolatos híreik, így az ötlet nem csak egy adott cégnél lehet helytálló. Ehhez elegendő a cég saját Rss feed-jének URI címét megadnia.

6.3 RSS feed technológia

Az RSS egy webes tartalmat szolgáltató hírfolyam, amely az XML 1.0-nak eleget tevő lényegében XML tartalom [8].

Az RSS web „szindikáció” segítségével lehetővé teszi, hogy egy weboldal tartalma több más weboldal is automatikusan megkapja, szinkronizálja [9]. A standard XML fájlformátum biztosítja a kompatibilitást más feldolgozó egységekkel/programokkal. A böngészőt utasíthatjuk arra, hogy automatikusan letöltse az adatokat a felhasználó számára [10].

A fájl struktúráján legfelsőbb szintű eleme az <rss> tag, amelynek kötelező attribútumának neve version, ez jelzi az RSS verzió számát. Az <rss> tag alá van rendelve a <channel> tag,

amely a kommunikációs csatorna meta adatait írja le. A <channel> tag-en belül az XML file tartalmazhat <item> tag-eket. Az „item” általában egy „sztorit” reprezentál, ami lehet például egy hír. Az <item> tag-en belül további elemek lehetnek, olyanok, mint például hír címe: <title>, leírása: <description> és így tovább [hiv rss hiv doksi].

```

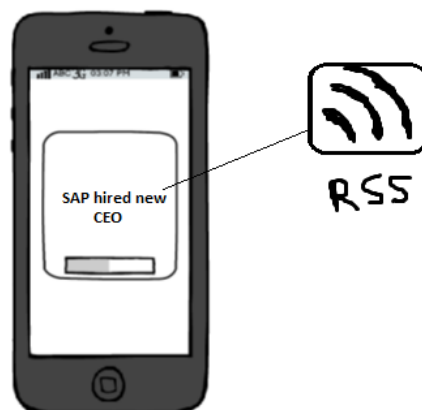
global.sap.com/corporate-en/press/newsroom/rss.aspx
This XML file does not appear to have any style information associated with it. The document tree is shown below.
<?xml version="1.0" encoding="UTF-8" ?>
<rss version="2.0">
  <channel>
    <title>SAP - Press Room</title>
    <link>
      http://global.sap.com/corporate-en/press/newsroom/rss.aspx
    </link>
    <description>Press Releases</description>
    <language>en-US</language>
    <copyright>Copyright 2011 SAP AG. All rights reserved.</copyright>
    <webMaster>webmaster@sap.com</webMaster>
    <lastBuildDate>Wed, 23 Oct 2013 00:00:00 GMT</lastBuildDate>
    <image>
      <title>SAP - Press Room</title>
      <width>73</width>
      <height>36</height>
      <link>
        http://global.sap.com/corporate-en/press/newsroom/rss.aspx
      </link>
      <url>http://www.sap.com/global/images/saplogo.gif</url>
    </image>
    <item>
      <title>
        SAP Positioned as a Leader for Seventh Consecutive Year in Gartner Magic Quadrant for International Retail Core Banking 2013
      </title>
      <link>
        http://global.sap.com/corporate-en/news.aspx?articleID=21776
      </link>
      <pubDate>Wed, 23 Oct 2013 00:00:00 GMT</pubDate>
    </item>
  </channel>
</rss>

```

12. kép: A SAP hír portál RSS feed-jének URI címe és annak XML tartalma.

6.4 Tervezés

6.4.1 Mock up

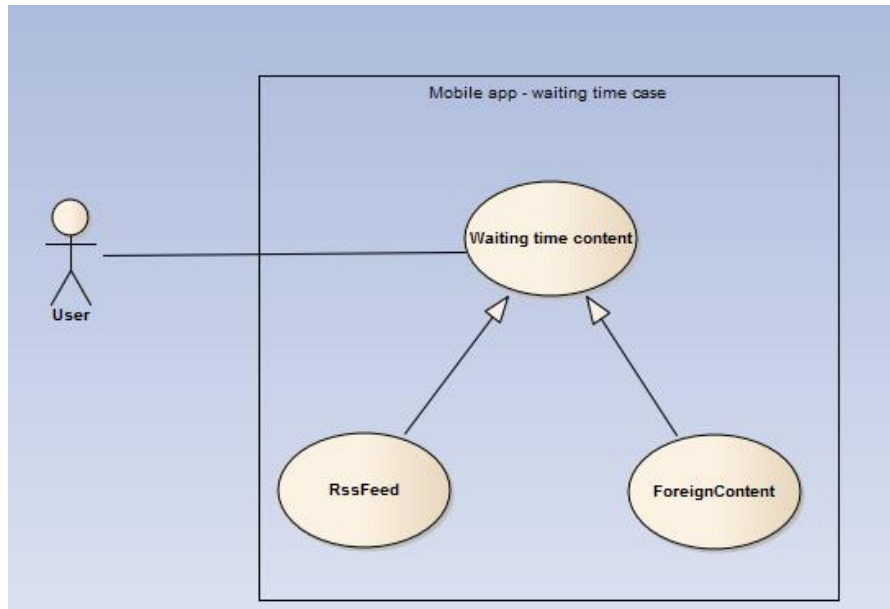


13. kép: Töltésidő Mock up-ja

6.4.2 Use-case diagram

Aktor: Töltésidő alatt unatkozó felhasználó.

Forgatókönyv: Olvassa az aktuális töltési idő alatt megjelenő tartalmat.



14. kép: Töltésidő use-case diagramja

6.4.3 Modell kidolgozás

6.4.3.1 Osztályok kidolgozása

BusinessAppActivity

Felelősség: Ez az osztály felelős a MyDialog osztály példányosításáért. Ezen az Activity-n belül történik a dialógus képernyőre rakása. Fontos, hogy átadja a kontextusát a MyDialog számára.

DialogContentManager

Felelősség: A dialógus megjelenítésekor többféle lehetséges tartalom elképzelhető. Itt történik például a külső programtól származó tartalom átvétele és továbbítása a MyDialog felé, amennyiben létezik ilyen és rá esett a választás. RSS szempontból feladata az RSS szolgáltatás kiválasztásának levezénylése valamint annak elindítása. A hírek címét továbbítja a MyDialog felé.

Ősosztályok:

nincs

Attribútumok:

resultReceiver

foreignDialogContent

items

foreign

Metódusok:

startService(): Elindít egy Rss szolgáltatást.

selectRss(): Kiválasztja, hogy melyik legyen a lehetséges Rss szolgáltatások közül az aktuális.

getDialogContent(): Az foreign boolean flag segítségével eldönti, hogy saját (RSS) vagy külső tartalomról van-e szó, ha például saját, akkor a selectRss() privát metódust meghívva kiválasztja melyik Rss szolgáltatást szeretné igénybe venni. Amennyiben külső programtól származó tartalmat szeretnénk fogadni és tesztelni, akkor a konstruktorban érkezik egy foreignDialogContent objektum és a foreign boolean flag automatikusan true értéket kap, így a getDialogContent() metódus a kívülről érkező tartalmat továbbítja.

MyDialog**Felelősség:**

Megkapja a kontextust az üzleti alkalmazástól, amit más osztályok számára elérhetővé tesz. Az örökölt setMessage() metódus paraméterének átadja az Rss hír címét.

Ősosztályok:

AlertDialog.Builder

Attribútumok:

context

Metódusok:

getContext(): visszaadja az Activity-től konstruktoron keresztül kapott kontextust.

getRssContent(): Az Rss címét String-kén szolgáltatja.

RssAdapter**Felelősség:**

Ha szeretnénk View-kat is feltölteni Rss tartalommal akkor az adapter teremt meg a Service és View közti kapcsolatot.

Ősosztályok:

BaseAdapter

Attribútumok:

items

context

Metódusok:

getCount(): A listában lévő View itemek számát adja vissza

getItem(): Visszaadja a View itemet.

getItemID(): A View item azonosítóját adja vissza.

getView(): Magát a View-t adja vissza.

RssCustomParser

Felelősség:

Az XML felépítésének megfelelően, az XML tag-ek segítségével értelmezi a bejövő szöveges adatforrást. Az RSS feed-ek XML tartalmainak felépítései eltérhetnek, amennyiben ez így van akkor külön Parsert-t kell készítenünk az eltérő XML feldolgozásához.

Őszosztályok:

nincs

Attribútumok:

nincs

Metódusok:

parse(): Visszaad egy RssItem-eket tartalmazó listát, amit a bemenetül kapott InputStream-ből (XML tartalomból) állít elő a readFeed() segítségével.

readFeed(): A parse metódus hívja meg, itt történik az RssItem-ek tényleges szétválasztása és előállítás. Az RssItem-eket egyesével egy listába pakolja minden egyes iteráció során.

readLink(): A link String változóba mentése és visszaadása.

readTitle(): A title String változóba mentése és visszaadása.

readText(): A nyitó és záró XML tag-ek közti eredmények kiolvasása.

RssItem

Felelősség:

Egy Rss válasz egy elemét (item) reprezentáló osztály.

Őszosztály:

nincs

Attribútumok:

title

link

Metódusok:

getTitle(): Visszaadja a címet.

getLink(): Visszaadja a linket.

RssService

Felelősség: Kapcsolatteremtés az adott szolgáltatással.

Ősosztályok:

IntentService

Attribútumok:

rss_link

items

reciever

Metódusok:

onHandleIntent():Elindítja a parse-olást.

getInputStream(): InputStream-et állít elő a megadott RSS kapcsolat megnyitása után.

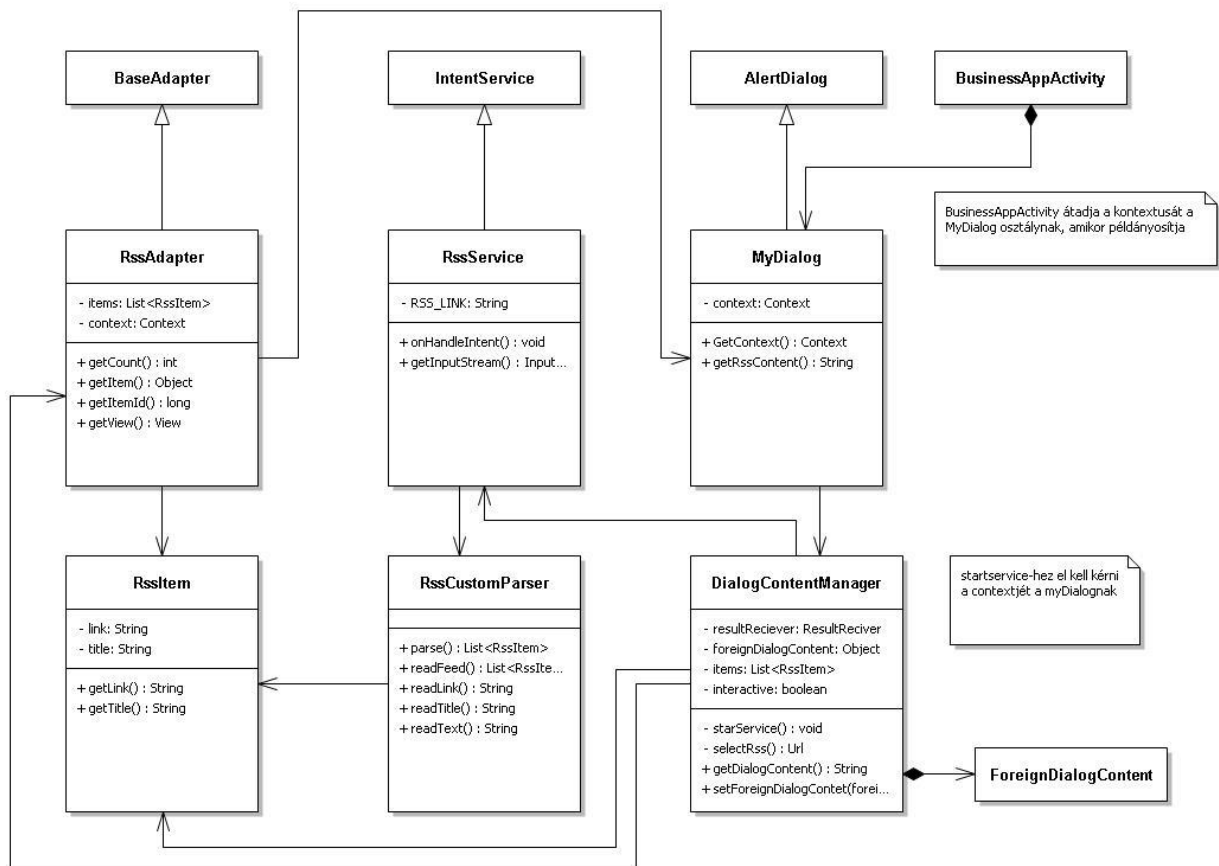
6.4.3.2 Osztálydiagramok

Áttekinthető, vázlatos osztálydiagram:



15. kép: Magasabb absztrakciós szintű diagram

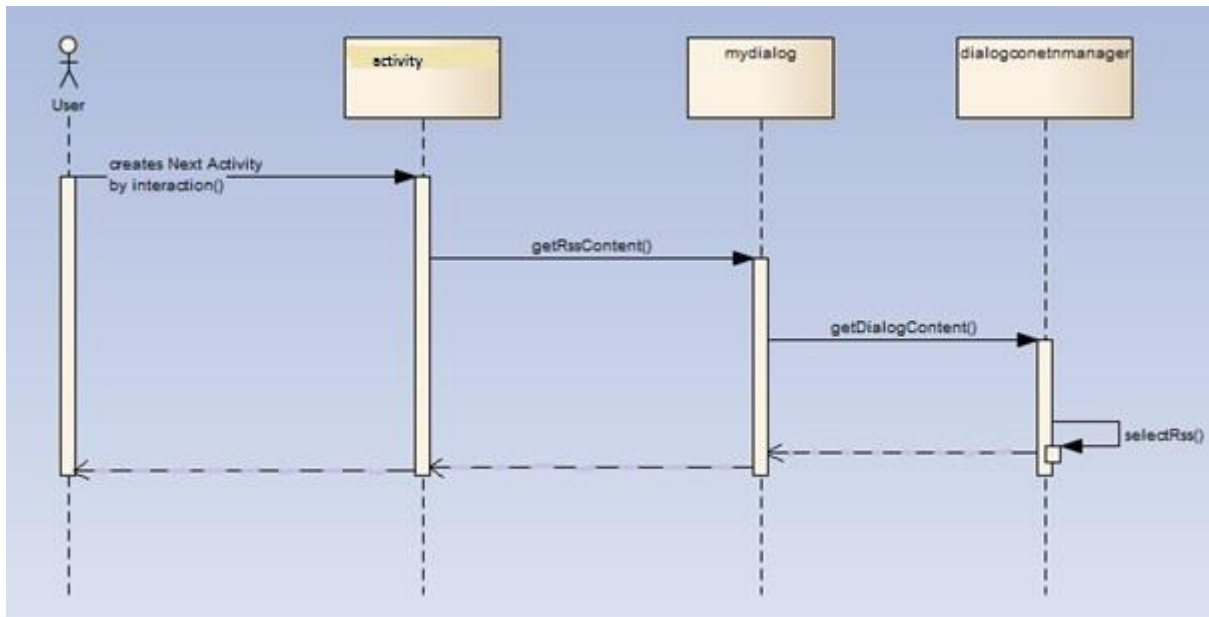
A szkeleton elkészítését elősegítő osztálydiagram:



16. ábra: Alacsonyabb absztrakciós szintű osztálydiagram

6.4.3.3 Szekvencia diagramok

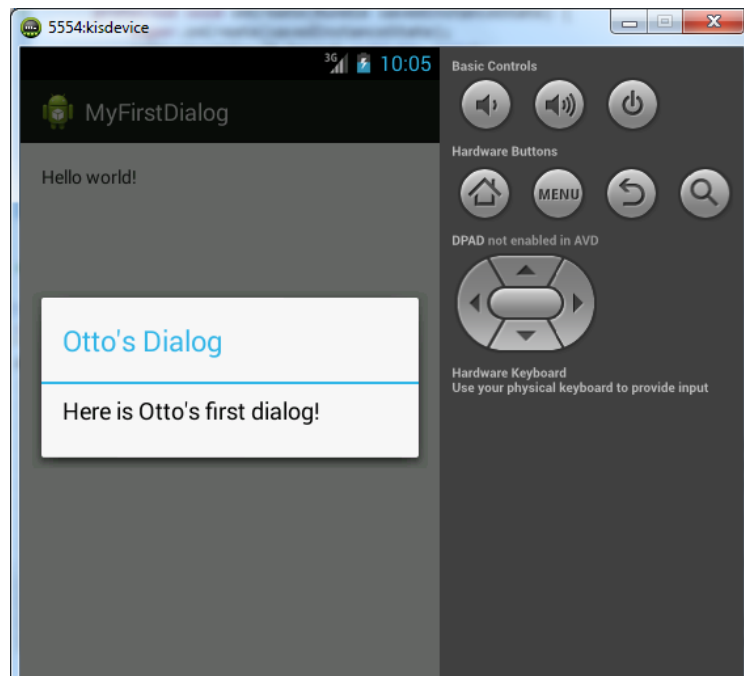
A felhasználó egy, a mobil eszközön végzett interakcióval új képernyő megjelenítését váltja ki létrehozva ezzel egy új activity-t. Amíg az alkalmazás következő funkcionális betöltődik, az aktuális activity példányosítja a MyDialog osztályt és elkéri tőle a töltési időben megjelenő dialógus tartalmát. A mydialog objektum a pontos dialógus tartalom beszerezése érdekében a dialogcontentmanager-hez fordul. A dialogcontentmanager elkéri a 17. ábrán nem szereplő Rss Service segítségével az RSS hírportál URI címének felhívására kapott XML tartalomból előállított értékeket. Vizsgálja, hogy pontosan milyen tartalmat továbbítson a mydialog felé. Végül a mydialog továbbítja az activity-nek a felhasználó számára töltési időben megjelenítendő dialógus tartalmát.



17. kép: A user-hez eljutó információ útját ábrázoló szekvencia diagram

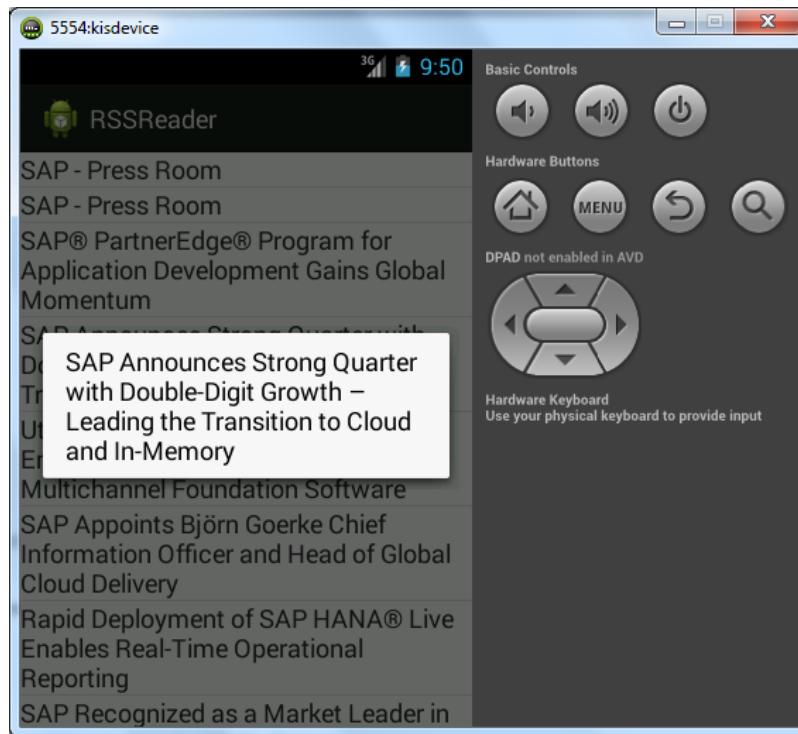
6.5 Szkeletonból prototípus

Miután megismerkedtem az Android fejlesztés alapjaival, megvizsgáltam hogyan lehet egy egyszerű dialógus ablakot készíteni. Az emulátor segítségével futatott program eredménye az x. ábrán látható.



18. kép: Az első Android dialógus

Tanulmányoztam az Android RSS feed kezelését, majd a fenti modell alapján létrehozott szkeleton metódusait implementáltam. A programot futtatva a következő eredményt kaptam:



19. kép: SAP – Press Room egy hírének címe dialógus ablakban

A fentiekben felvázolt architektúra alapján az integrálás lehetséges a HR Approvals és egyéb Android üzleti alkalmazásokba elősegítve ezzel a dolgozat elején megfogalmazott kérdésekre vonatkozó válaszok kutatását.

7. Felmérés és eredmények tervezése

7.1 Felmérés

A cél, hogy a fent említett alkalmazás egy valós üzleti rendszerbe való integrálása után odaadható legyen végfelhasználóknak és a véleményüket egy kérdőív keretein belül kifejtthessék.

A felmérés célközönsége egy részről azok az emberek, akikkel interjúkat készítettem (lásd melléklet). A program tesztelése után a kételkedő kollégák, mint például a „quality”-s vagy „product owner” vajon elmozdulnak-e pozitív irányba a „gemifikációval” kapcsolatosan? Illetve azok, akik kezdetektől láttak fantáziát a dologban, továbbra is így gondolják-e? Más részről a felmérés további célszemélyei az SAP fejlesztő kollégái, mérnökhallgatók, akik technikailag is az alkalmazás mögé látnak és ilyen szemmel is tudnak választ adni arra, hogy alkalmaznák-e ezt az ötletet. A program felhasználóiról sem szabad megfeledkeznünk, akik az üzleti alkalmazások kezelői. Őket érdemes szembesíteni egy szokásos homokóra animációval valamint néhány töltésidőben megjelenő passzív, dinamikus (ez lehet passzív is) vagy interaktív tartalommal és kideríteni melyik felelteti vele a töltési időt rövidebb illetve hosszabb távon.

7.2 Eredmények

Eredményként egyértelmű tényeket kell közölni a felmérés és a témában való kutatás segítségével. Ilyen jelenleg fiktív eredmények pl.:

- **A felhasználó maximum 20 másodpercet hajlandó várni, ennél nagyobb töltési idő esetén letörli az alkalmazást.**
- **A töltési idő alatt elhelyezett „gemifikációnak” akkor van értelme, ha ez az időtartam legalább 2, legfeljebb 15 másodperc. Különben a felhasználó törli a „gemifikációt”, mert vagy nincs ideje feldolgozni az információt vagy a túl sokáig látja azt.**
- **A felhasználók hajlandóak több időt várni, annak érdekében, hogy számukra hasznos adatot olvassanak.**
- **7 másodpercnyi várakozást a felhasználók átlagosan 5,3 másodpercnek érznek, ha homokóra helyett számukra hasznos információt jelenítünk meg.**
- **Az interaktív tartalom rövidtávon felelteti a töltésidőt, de hosszabb távon a felhasználók kikapcsolják.**
- **A fejlesztendő szoftverek tesztelésének és minőségellenőrzésének vezetője szerint, ha a „gemifikáció” plusz két másodpercnél többel növeli a töltésidőt, akkor az integrálása kerülendő.**

A dolgozat maga és az eredmények közzéte mának kell, hogy szolgáljon olyan fejlesztők számára, akik üzleti alkalmazáson belül szeretnék a töltési időt kihasználni az ekkor megjelenő tartalom „gemifikálásával”.

Irodalomjegyzék

A hivatkozások egy része csak azon olvasók számára elérhető, akik rendelkeznek hozzáféréssel az SAP belső hálózatához.

[1] Gamification Wikipédia: <http://en.wikipedia.org/wiki/Gamification>

[2] „The Game of Work” könyv: <http://www.goodreads.com/book/show/3476263-the-game-of-work>

[3] SAP gamification applications: <https://community.wdf.sap.corp/docs/DOC-37661>

[4] SCN fórumon általam feltett kérdések a ”gemifikációról” töltésidő közben: <https://community.wdf.sap.corp/message/450170#450170>

[5] Fiori problémák:

<https://wiki.wdf.sap.corp/wiki/download/attachments/1423055092/Kapsel+Browser+-+Architecture+Blueprint.pdf?version=3&modificationDate=1376642390347>

[6] Fiori alap működés: <http://help.sap.com/fiori>

[7] Várakozási idő pszichológiája: <http://blog.usabilla.com/how-to-integrate-waiting-time-user-experience/>

[8]RSS hivatalos honlap: <http://cyber.law.harvard.edu/rss/rss.html>

[9]Web syndication: http://en.wikipedia.org/wiki/Web_syndication

[10]RSS Wikipédia: <http://en.wikipedia.org/wiki/RSS>

Terminológia

Magyar jelentés	Angol jelentés	Angol rövidítés (ha van)	Definíció
Activitiy	Activitiy		Android környezetben azt a dolgot definiálja amit éppen a felhasználó tenni tud
Anyagszükséglet tervezés	Material Requirement Planning	MRP	Anyagszükséglet tervező modul
backend	backend		Üzleti logikát tartalmazó rendszer
felhasználói interfész	UI		felhasználói felület.
gemifikáció	gamification		Lásd 3. fejezet
HANA	HANA		SAP adatbázis kezelési módszer
hibakeresés	debug		Hibakeresés a kódon belül
keretrendszer	framework		valamely példányt (lehet akár az egész futó program) kezelő, futtató rendszer
meta adat	meta data		adateleírás
Mock up	Mock up		Kézzel rajzolt, funkcionalitást és nem UI-t bemutató pillanatkép az alkalmazásról
SAP Corporate portál	SAP Corporate portal		Lényegében Kezdőlap a SAP dolgozói számára, fontos hírek, beállítások érhetőek el innen
SAP Kommunikációs Hálózat	SAP Community Network	SCN	SAP-s fórum és Wikipédia tartalom.
SRM	Supplier Relationship Management	SRM	Eszközbeszerzést segítő SAP alkalmazás
start up	start up		Egyedi ötletből kiinduló innovatív vállalkozás

szkeleton	skeleton		Az osztályoknak csak a metódusait és tagváltozóit tartalmazó kód, a metódusok implementálása nélkül
töltés idő	loading time		lásd 5. fejezet
töréspont	breakpoint		A futó programot a töréspont sorában megállítja
törzs felhasználó	heavy user		Olyan felhasználók akik komplex üzleti funkciókat használnak
tutorial	tutorial		A felhasználó virtuális végigvezetése az alkalmazás egyes funkcióin
URI	Uniform resource identifier	URI	Böngészőbe beírt cím. Az URL ennek egy speciális változata
Web „szindikáció”	Web syndication		Lényegében webtartalom szinkronizációja
webszerver	web server		HTTP feletti kéréseket fogadó és válaszokat küldő szerver

Mellékletek

1. melléklet:

Típus: Interjú

Megkérdezett: Novoth Péter

Beosztás: SAP terméktámogatói csoportot vezető menedzser

Téma: Töltés idő alatt elképzelhető Menedzser specifikus adatok

Péter nem találkozott még a HR Approvals nevű alkalmazással ezért először funkcióin keresztül bemutattam az alkalmazást.

Arra kértem, hogy gondolkozzon olyan információkon, amelyek neki, mint menedzsernek hasznosak lehetnek és néhány másodpercben felfoghatóak. Véleménye szerint bármely kisebb pop-up-ban megjelenített szöveges adat feldolgozható és memorizálható akár egy-két másodperc alatt is. Ezt alátámasztva a villámolvasást hozta fel példának.

Ötletként hirtelen a céges részvények árfolyamának megjelenítését hozta fel, de azt is hozzá tette, hogy számára hasznos információk megjelenítése érdekében, hajlandó lenne néhány másodperccel többet várni. Például beáldozná hajlandó lenne elfogadni a megnövelt töltési időt, ha a 'Leave Requests' menüpont érintése után véletlenszerűen a kritikus időpontokban (pl. karácsony) szabadságon lévő csapattagok számát jelenítenék meg.

2. melléklet:

Típus: Interjú

Megkérdezett: Horváth-Dori Gergely

Beosztás: SAP informatikus – technikai terméktámogató

Téma: Töltésidő alatt elképzelhető tartalmak, RSS feliratkozás

Gergő alapfeladat mellett, a cég kávézójában lévő okos TV-k tartalmát szerkeszti. Ezeken tipikusan olyan információk vannak, amelyek egy adott beszélgetés közben rápillantva is értelmesek tudnak lenni: ilyenek például a cég dolgozói által üzleti úton készített képek, céges eredmények, részvények állása, céges hírek.

Gergőt azért kerestem meg, hogy megkérdezzem véleményét a töltésidő kihasználásáról illetve számítottam az RSS feed-ekkel kapcsolatos tanácsaira is.

Gergő szerint jó ötlet ilyen adatokban gondolkozni töltésidő alatt. Rámutatott a három Rss feed-re; az időjárás, az SAP részvény árfolyam és az SAP Corporate portál híreinek RSS URI címeire.

3. melléklet:

Típus: Interjú

Beosztás: SAP Fejlesztő -Product owner

Téma: Gemifikáció az üzleti alkalmazások területén.

Az interjúalany hétköznapi munkájának része a fejlesztésekkel kapcsolatos ötletek, követelmények megfogalmazása. Az Ő szava roppant mérvadó a témában, hiszen végső soron neki kell rendelkezni arról, hogy például egy elkészült „gemifikációs” megoldás belekerüljön-e a programba.

A megkérdezett Product owner véleménye a „gemifikációval” kapcsolatosan, hogy az általa indítványozott és engedélyezett fejlesztések célközönségének nehezen elképzelhető. A szoftvereiket „heavy user”-ek alkalmazzák (ilyenek például a beszerző specialisták, elemzők), akik nap-nap ugyan azokat a komplex üzleti funkciókat használják. Szerinte számukra a „gemifikáció” hátráltató és idegesítő tényező lenne egy idő után. Rámutatott azért arra is, hogy még ilyen szoftverek esetén is akad olyan hosszú várakozási idő (pl. Anyagszükséglet tervezés - MRP futtatása), ahol érdemes lenne elgondolkodni a felhasználó „gemifikálásáról”. Azonban szerinte itt is számolni kell azzal, hogy ha túl hosszú a várakozási idő, akkor valószínűleg a felhasználó inkább megnyit egy böngészőt vagy iszik egy kávé.

4. melléklet:

Típus: Rövid eszmecsere

Megkérdezett: Tatár Péter

Beosztás: SAP Fejlesztő –Quality menedzser

Téma: Gemifikáció az üzleti alkalmazások területén.

Péter munkája során teszteseteket ír ki és minőségi szempontból vizsgálja a fejlesztés alatt álló szoftvereket. Vele csak néhány szót volt alkalmam váltani.

Véleménye szerint nem támogatná a „gemifikációs” megoldásokat, hiszen azok rontanák az alkalmazások teljesítményét, sebességét.