EGYSÉGES KÖZBESZERZÉS FIGYELŐ RENDSZER ARCHITEKTURÁLIS ÉS FUNKCIONÁLIS TERVEZÉSE

Tartalom

1. Bevezető 4
   1.1. A dokumentum célja és határai 4
   1.2 A rendszer ismertetése 4
   1.3 Adatlap 4
2. Architektúra 5
   2.1. Főbb irányelvek 5
   2.2. Az architektúra felépítése 6
   2.3. Informational context (C1) 6
   2.4. Containers (C2) 7
   2.5. Component (C3) 8
3. Funkcionális követelmények 9
   3.1. Felhasználók 9
      3.1.1. Anonim látogatók [CMS]: 9
      3.1.2. Regisztrált felhasználók (tesztelők, előfizetők, passzív felhasználók) [APP]: 9
      3.1.3. Marketinges [CMS, EDM] 10
      3.1.4. Content Manager [CMS] 10
      3.1.5. Call centeres [APP, CRM] 10
      3.1.6. Customer support [APP, CRM] 11
      3.1.7. Adminisztrátor [CMS, APP, CRM, EDM] 11
   3.2. Marketing csatornák 11
   3.3. Crawler 12
   3.4. Letöltés 12
   3.5. Feldolgozás 13
   3.6. Indexelés 14
   3.7. Találatok generálása 14
   3.8. Találat értesítő e-mailek 14
   3.9. Ütemezés 15
4. Alkalmazás (APP) 15
      4.1. Regisztráció / Bejelentkezés 15
      4.2. Beállítások 15
      4.3. Keresés 16
      4.4. Keresőprofil 17
      4.5. Keresési eredmények exportálása 17
      4.6. Keresési eredmények kiküldése e-mailben 18
      4.7. Találatok megtekintése 18
      4.8. Hirdetmények megtekintése 18

4.9. Hirdetmény letöltés....19
4.10. Kedvencek....19
4.11. Olvasottnak jelölés....19
4.12. Hirdetmények követése....19
4.13. Kiírók és nyertesek megjelenítése....20
4.14. Kiírók és nyertesek követése....20
4.15. Előfizetések menedzselése....20
4.16. Dashboard....21
4.16.1. [admin] Előre definiált keresőprofilok....22
4.16.2. [admin] Offline hirdetmények feltöltése....22
4.16.3. [admin] Hirdetmények szerkesztése....22
4.16.4. [admin] Más user nevében bejelentkezés....23
4.16.5. [admin] Admin felület....23
4.16.6. [admin] Statisztika....23
4.16.7. [admin] Levél sablonok szerkesztése....24
4.17. Többnyelvűsítés (i18n)....24
4.18. API....24
5. Marketing oldal (CMS)....25
5.1. Alap CMS funkcionalitások....25
5.2. Webshop....25
5.3. Közbeszerzés figyelő modul....26
6. Ügyfél adatbázis (CRM)....26
6.1. Ügyfelek menedzselése....27
6.2. Szolgáltatások managelés....27
6.3. Termékek / Előfizetések menedzselése....27
6.4. Automatikus előfizetés hosszabbodás....28
6.5. Taskok menedzselése (call center funkciók)....28
6.6. Hibabejelentések kezelése....28
6.7. Levél küldés....28
6.8. Figyelmeztetések....29
6.9. Statisztika....29
6.10. Export / Import....29
6.11. EDM (integráció)....29
6.12. Webshop (integráció)....29
6.13. API....30
6.14. Többnyelvűség (i18n)....30
6.15. Jogosultság, láthatóság....30

6.16. Affiliate rendszer kialakítása....30
7. EDM....30
7.1. Felhasználói csoportok....31
7.2. Felhasználói adatok....31
7.3. Szűrők....31
7.4. Levél sablonok....31
7.4.1. Ütemezés, levél küldés....31
7.5. API....31
7.6. Analitika....32
7.7. Integráció....32
7.8. Monitorozás....33
8. Elosztott működés....33
9. Nem funkcionális követelmények....33
MELLÉKLETEK....35
I melléklet: C1 diagram....36
II. melléklet C2 diagram:....37
III. melléklet: Engine alrendszer kibontása:....38

1. Bevezető

1.1. A dokumentum célja és határai

Jelen dokumentum célja az egységes közbeszerzési figyelő rendszer architekturális tervezésének főbb irányvonalainak ismertetése, tervezési döntések magyarázata, átfogó kép bemutatása a rendszer működéséről, funkcióiról.

A dokumentum célközönsége a fejlesztésben résztvevő és az üzemeltetésért felelős emberek.

1.2 A rendszer ismertetése

A rendszer célja az ügyfelek informálása napi szinten megjelenő közbeszerzési hirdetményekről. Az online elérhető felületekről napi rendszerességgel letölti a hirdetményeket, azokat transzformálva kereshető állapotba teszi elérhetővé a kliensek preferenciái alapján.

A kliensek keresőprofilok definiálásával adhatják meg a keresési feltételeiket, amire illeszkedő hirdetményeket napi rendszerességgel e-mailes értesítő formában kapnak meg. A szolgáltatás nemzetközi piacra lett tervezve, így annak képesnek kell lennie több ország hirdetményeket megjelenítő portálját követni. Mindemellett fontos szempont a rendszer tervezésénél a skálázhatóság, robusztusság, magas elérhetőségi ráta, minden olyan szempont, ami egy mai modern webes alkalmazással szemben elvárásként felállított.

1.3 Adatlap

Adatlap	
Software környezet	Java SE 8.0, illetve IE 9+, Firefox 2+ és Chrome 17, Ubuntu 10.04+
Hardware környezet	A rendszer virtualizált környezetben fog futni, az erőforrások nem lesznek transzparensek
Architektúra	N- rétegű (Frontend, Backend, Engine, Adatbázis), illetve egyéb, kiegészítő rendszerek (CRM, EDM)
GUI	Vékony-kliens
Adatbázis	Relációs adatbázis: MySQL, Dokumentum-orientált: MongoDB, Kereső-szerver/Indexelő: Elastic Search
Szerver	Apache Tomcat 8+ illetve J2SE alapú alkalmazások
Program és scriptnyelvek	Java, HTML, Javascript(jQuery), SQL

2. Architektúra

2.1. Főbb irányelvek

Az architektúra tervezése során az alábbi irányelvek követése volt szem előtt:

- **Robusztusság - hibatűrés**: a rendszer komponensei egymástól függetlenül legyenek működőképesek, és az egyes komponensek hibája, leállása ne hasson ki a további komponensek működésére. A megvalósítás során cél az állapotgép-típusú működés, illetve a perzisztens üzenet-tárolás (azaz egyes komponensek egymással főként adatbázison keresztül kommunikálnak)

- **Közös platform**: a rendszer sok, kisebb komponensből áll, azonban karbantarthatósági szempontok miatt ahol lehetséges, mindenképpen azonos programozási nyelv használata szükséges (mind kliens, mind szerver oldal esetén) ezzel csökkentve a hibaelhárítás időigényét, az üzemeltetéshez szükséges tudásbázissal együtt

- **REST-API használata/Micro services**: minden esetben a rétegek közötti kommunikáció REST protokollon keresztül történik – ide értve a frontendet és backendet - ezzel is növelve az egyes modulok cserélhetőségét, párhuzamos fejleszthetőségét, és így elősegítve a minél lazább csatolásukat. Az egyes modulok közötti ütemezést egy közös ütemező látja el, aminek feladata a folyamatok elindítása lesz.

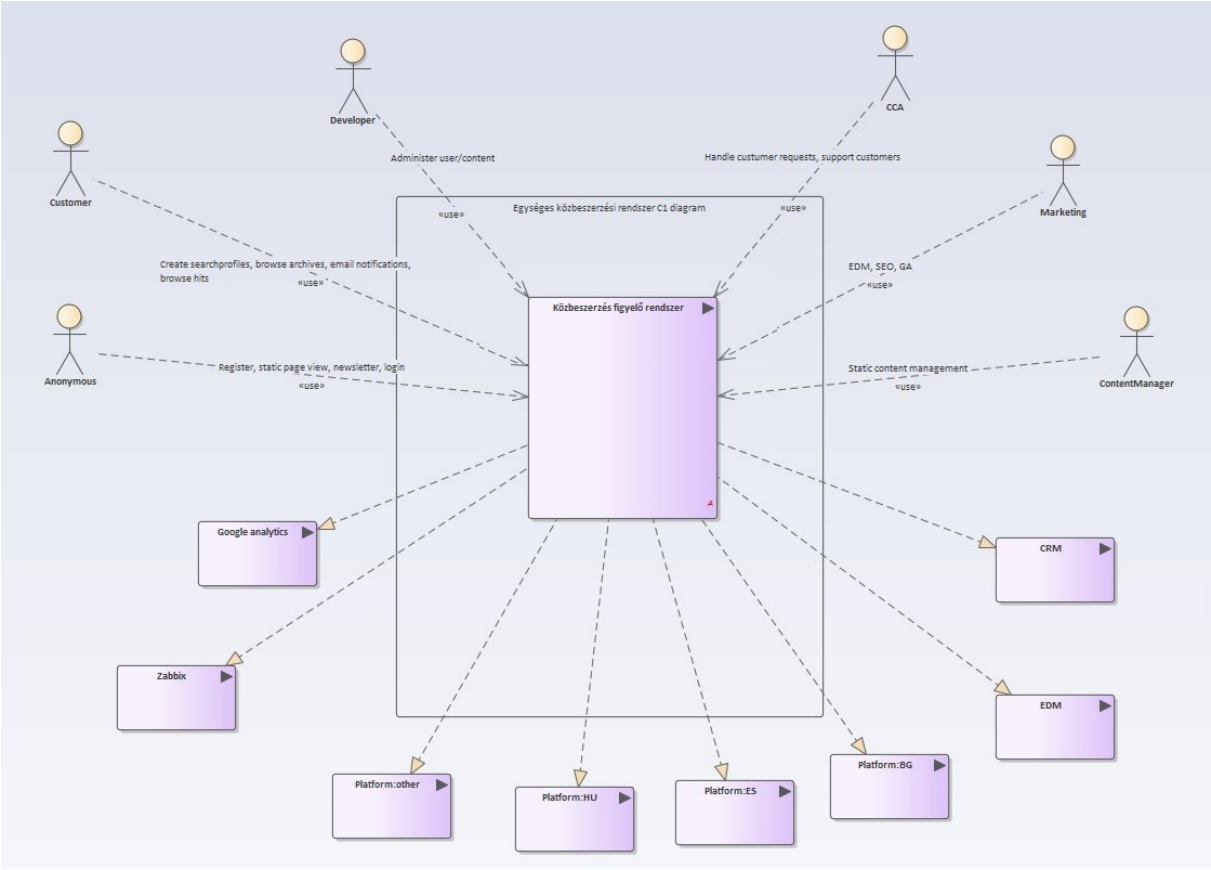
- **Konfigurálhatóság**: mivel a rendszer életútja során újabb és újabb platformok bevezetése tervezett országonként, ezért az alkalmazásnak támogatnia kell ezt a folyamatot oly módon, hogy az újabb bevezetések a lehető legkevesebb kód írásával, inkább konfigurációval lehessen megoldani. Ez alatt az ország specifikus részek kiemelését és konfigurálását értjük.

2.2. Az architektúra felépítése

Az architektúra felépítésének leírásához C4 modellt használjuk, mellyel a rendszer nagy egységeitől kiindulva a konténer rétegig mutatjuk be annak struktúráját.

2.3. Informational context (C1)

A rendszer C1 diagrammja (I. melléklet) az alábbi képen tekinthető meg:



Az ábrán feltüntetésre került a központi egységet alkotó Közbeszerzés figyelő rendszer, a rendszer felhasználói és a rendszerhez kapcsolódó alrendszerek.

Felhasználók tekintetébe jogosultságok alapján válnak elérhetővé a rendszer funkciói. Jogosultságot regisztráción keresztül, illetve előfizetési csomag megvásárlásával lehet szerezni. E nélkül a rendszer statikus tartalmai, a regisztrációs felület és a hírlevél feliratkozás érhető csak el anonymous felhasználó számára. Regisztrált felhasználóként aktív előfizetéssel lehetőség van keresőprofilokat beállítani, archív hirdetmények között keresni, találatokat megtekinteni, újabb előfizetést kérvényezni.

A rendszer üzemeltetését, karbantartását, illetve a tartalmak menedzselését külön, a rendszerben definiált szerepkörök szerint más-más felhasználóknak kell kezelni. Ezek közül CCA az, aki az ügyfelek megkereséseit kezeli közvetlenül, illetve rajta keresztül jutnak el a hibák a fejlesztőkhöz, akik kezelik ezeket. Külön szerepkörrel érhető el a statikus tartalmak

kezelése és a marketinges tevékenységek, úgy, mint az oldal SEO-zása, követő kódok analitikája, marketing levelek kezelése.

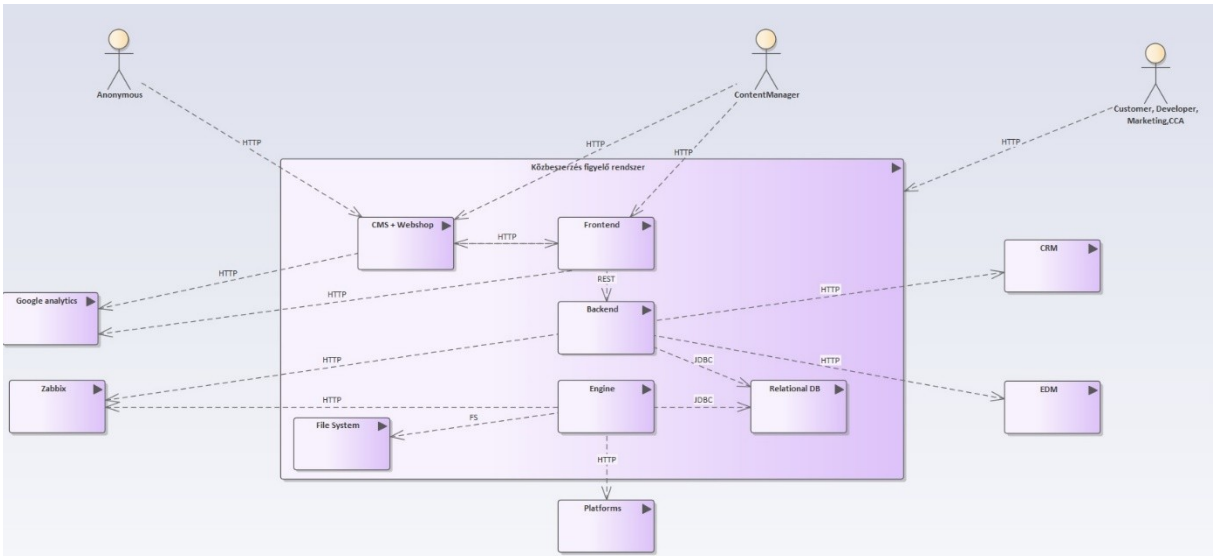
Az ábra alsó szegmensébe lettek elhelyezve azok az alrendszerek, melyek kapcsolódva a Közbeszerzési figyelő rendszerhez adatokat tárolnak további szolgáltatások kiszolgálásához.

Ezek az alábbiak lesznek:

- **Google Analytics**: külső rendszer, amely analitikai (naplózási) adatokat tárol és menedzsel
- **Zabbix:** külső rendszer, amely monitorozza automatikusan a rendszer terheltségét, elérhetőségét, erőforrás felhasználását (hálózati elérések, CPU/RAM/HDD felhasználás, elérhetőség, JMX, stb.)
- **CRM:** Customer Relation Management rendszer; a felhasználói adatainak, illetve kapcsolódó információknak (például: előfizetések) tárolása
- **EDM:** Electronic Direct Marketing rendszer; nagy mennyiségű (hírlevél-típusú) e-mail-ek küldésére szolgáló rendszer
- **Platform:** a rendszer által figyelt, közbeszerzési hirdetményeket megjelenítő oldal(ak)

2.4. Containers (C2)

A rendszer C2 diagrammja az alábbi képen látható (II. melléklet):



Az ábrán jól kivehető, milyen konténerek futnak és azok milyen protokollon keresztül kommunikálnak egymással. Rendre az alábbiak szerint lett kialakítva a rendszer:

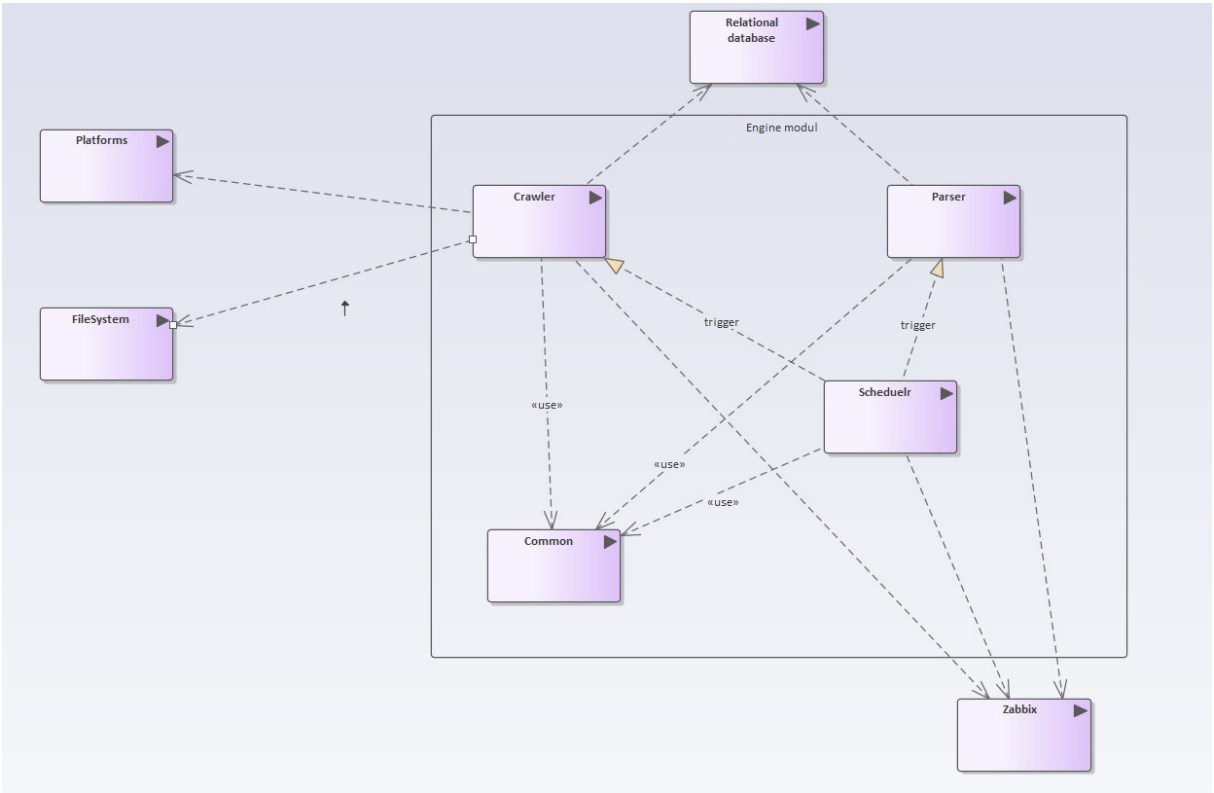
- **CMS + Webshop rendszer**: Content Management System, amely a gyakran változó, statikus oldalakat tartalmaz (például: hírek); az implementáció alapja Wordpress, illetve LAMP stack. A webshop része felelős az előfizetések kezeléséért. Minden előfizetés, ami időtartamban és keresőprofil számban tér el, mintegy termék lesz megvásárolható a webshopon keresztül.
- **Frontend**: a felhasználók által elérhető központi, bejelentkezést igénylő felület; az alatta futó konténer statikus fájlszerver (Apache) illetve Angular JS implementáció

- **Backend**: a frontend üzleti logikájának megvalósítása; Apache Tomcat webszerver és Spring MVC alkalmazás (Java 8)
- **Engine**: az ütemezett feladatok (letöltés, értesítések, stb.) felelős al-rendszer; Spring-alapú modulok (Java 8) több, különböző JVM-ben futva
- **Database**: a rendszerben található statikus, relációs és dokumentum-orientált tárolásért felelős komponense (MySQL és MongoDB)
- **File System**: az adatbázisban tárolandó, nagy méretű fájlok (például: pdf hirdetmények) által igénybe vett fájlrendszer

Az egyes modulok/komponensek közötti kommunikáció minden esetben http(s)-en keresztül zajlik.

2.5. Component (C3)

A komponensek közül az Engine alrendszer kibontását tartalmazza ez a diagram (III. melléklet). Ez a rendszer felelős a hirdetményeket megjelenítő oldalak követéséért. Napi szinten töltik le az újonnan megjelent közbeszerzési hirdetményeket és alakítják át a megfelelő struktúrába, amit aztán egy másik alrendszer feldolgoz. Az alábbi ábrán látható ennek az alrendszernek a felépítése:



A diagramról jól leolvasható, hogy az Engine alrendszer 4 nagyobb modulból épül fel, melyek rendre:

- **Crawler:** feladata a platformokról letölteni a hirdetmények html tartalmát
- **Parser:** a letöltött html-ből nyeri ki a hirdetmény adatait
- **Scheduler:** ahogy a neve is következik az ütemezésért felelős modul

- **Common:** mindhárom modul ebből a modulból indul ki. A közös részeket foglalja magába, úgy, mint adatstruktúra, konverterek.

3. Funkcionális követelmények

Ebben a részben a rendszerrel támasztott elvárásokat szedtük pontokban. A rendszer funkcióinak egy része jogosultság nélkül érhető el, míg más funkciói jogosultsághoz kötött. Minden alrendszer definiálja a saját szerepköreit, s a hozzá tartozó elérhető funkciókat. A listába szedett funkciók ezek mentén, alrendszerenként készültek el. Az ügyfél közvetlenül csak az alkalmazás (APP) részével került interakcióba az egész rendszernek. A többi alrendszer csak az adminisztrátoroknak és menedzsereknek lesz elérhető.

3.1. Felhasználók

3.1.1. Anonim látogatók [CMS]:

Ezek a felhasználók azok, akik nem regisztráltak még be az oldalra, tehát nem szereztek semmilyen szerepkört. Számukra a statikus tájékoztató oldalak érhetőek el, illetve előre definiált keresőprofilok találatait tekinthetik meg.

- CMS Statikus oldalakat néznek meg
- Előre definiált keresések (NUTS, CPV) böngészése
- Hírek olvasása
- Regisztráció

3.1.2. Regisztrált felhasználók (tesztelők, előfizetők, passzív felhasználók) [APP]:

A regisztrációhoz a felhasználónak a belépéshez szükséges felhasználó nevén kívül (a jelszó automatikusan generált, az aktiválást követően az első bejelentkezés után lehet megváltoztatni) meg kell adni a számlázási adatait, amit a rendeléshez használ fel a rendszer a számla kiállításához. Ezen kívül itt kell megadni a preferált érdeklődési körét, ami alapján a rendszer le is generálja az első keresőprofilját a felhasználónak. Minden regisztráció egy 14 napos tesztidőszakkal veszi kezdetét, ami azt jelenti, hogy ebben az időszakban ingyenesen korlátlan számú keresőprofilt állíthat be, s ezekről napi szinten kapja az e-mailes értesítőt a keresési feltételeknek megfelelő hirdetményekről.

- Keresőprofilokat hoznak létre
- Archív keresést végeznek
- Kedvencek közé raknak hirdetményeket
- Hirdetmények részletes adatainak megtekintése
- Hirdetmények letöltése, exportálása
- Találatértesítők e-mailben

3.1.3. Marketinges [CMS, EDM]

Ahhoz, hogy a kereső enginek minél jobb besorolást adjanak az oldalakra, biztosítani kell az oldalak metázhatóságát, tartalmának formázhatóságát, amihez egy CMS rendszer kész megoldást ad. Ehhez kapcsolódik a marketinges megkereséseket segítő analítikák előállítása, illetve EDM kampányok küldésének biztosítása, amit külön erre beemelt EDM rendszer végez majd.

- SEO-t végez a CMS oldalon (tartalomírás megfelelő kulcsszó sűrűséggel, metadatok beállítása, szép url-ek, stb)
- Adwords kampányokat hoz létre, azokat optimalizálja
- Google analytics statisztikákat elemez
- Webshop statisztikákat elemez
- EDM kampányokat tervez és szövegez

3.1.4. Content Manager [CMS]

A statikus tartalmak egyszerű, gyors módosíthatóságához lett CMS rendszer beemelve, aminek segítségével az oldal könnyen testre szabható.

- Tartalommal tölti fel a marketing oldalt (CMS)
- Menüstruktúrát tervez és rak össze
- Adott nyelvre lefordítja az oldalt és szövegeket fogalmaz meg
- Híreket tölt fel az oldalra
- Karbantartja az oldal tartalmi részeit
- Bannereket és egyéb widgeteket helyez el az oldalra

3.1.5. Call centeres [APP, CRM]

Az ügyfelek megkereséséhez szintén egy külön alrendszerre támaszkodva kapható támogatás. A CRM alrendszer feladata az ügyfelek nyomon követése, ami alapján a leginkább relevánsabb ajánlatokat teheti meg a call centeres. Ehhez a rendszer képes rögzíteni az ügyfél adatait, tevékenységeket tud hozzájuk rendelni, úgy mint e-mailes megkeresés, telefonos megkeresés. Ezen kívül statisztikai adatokat szolgál a rendszerben rögzített adatok alapján.

- Hideghívásokat hajt végre, próbál új ügyfeleket akvirálni
- Hívólistákat szerez és állít össze

- Ügyfelekkel kapcsolatos adatokat rögzít
- Teszt ügyfeleket regisztrál
- Teszt ügyfeleknek keresőprofilt állít be

3.1.6. Customer support [APP, CRM]

Az ügyfelek megkereséseinek kezelésére a rendszernek felületet kell biztosítani az adatok módosításához, illetve a felhasználók hibabejelentéseinek rögzítéséhez. Ehhez szintén a már korábban említett CRM rendszer szolgáltatásait az alkalmazás (APP) funkcióival egészítjük ki.

- Bejövő hívásokat, panaszokat kezel
- Hibajegyeket rögzít
- Ügyfeleknek keresőprofilokat állít be
- Ügyfél adatokat kezel
- Előfizetéseket kezel

3.1.7. Adminisztrátor [CMS, APP, CRM, EDM]

Adminisztrátor feladata az ügyfelek kezelése, azok jogosultságának módosítása. Ehhez minden alrendszert el kell érnie és ott minden művelet végrehajtásához rendelkezik a megfelelő jogosultsággal.

- Bármely rendszerbe be tud lépni
- Rendszerszintű beállításokat tud elvégezni
- Support Level 3/4-ként ügyfél panaszokat kezel

3.2. Marketing csatornák

Ahhoz hogy a közbeszerzés figyelő rendszer széles közönséget megtaláljon, a marketinges szakembereknek minden féle csatornán keresztül el kell tudni érni a célközönséget. Ehhez az alábbi eszközök biztosítják a minél nagyobb felhasználó bázis kiépítését:

- Google természetes találatok
- Google adwords

- EDM kampányok
- Partner oldalakon elhelyezett reklámok, egyéb tartalmak
- Hideghívás
- Rendezvények
- Ajánlás

3.3. Crawler

A platformokon megjelenő hirdetmények letöltéséhez egy háromlépéses algoritmus lett megalkotva, aminek a végrehajtását követően elő állnak a letöltött, áttranszformált hirdetmények, amik továbbítható formában lesznek az indexelőnek. Azért lett a folyamat szétbontva, hogy párhuzamosan lehessen futtatni ezeket, így gyorsítva a hirdetmények előállítását. További lépéseket már azért nem lehetet beszúrni, mert ez már nem csökkentené szignifikánsan a futási időt.

Az első lépésben az összes olyan hirdetmény url-jét, ami még nem lett korábban letöltve összegyűjti a rendszer, majd az url alapján betölti a hirdetmény oldalát és letölti annak teljes html tartalmát. Ezt a html tartalmat fogja feldolgozni végezetül a rendszer. Az első feldolgozó mindig egyben tölti le az új hirdetményeket, míg a másik kettő a hirdetmény aktuális státusza alapján mindig egyet fog feldolgozni. Pont emiatt ez a két feldolgozó sokkal sűrűbben fog lefutni napi szinten.

3.4. Letöltés

Ennél a feldolgozásnál az elsődleges szempont az volt, hogy legyen a lehető leguniverzálisabb, azaz mindegyikére (vagy legalább is a legtöbbre) legyen felkészítve. Ez alapján az alábbiaknak kell megfelelnie ahhoz, hogy a letöltés gyors legyen, a változásokra a legkevésbé legyen érzékeny és egy újabb letöltő elkészítése a lehető legrövidebb idő alatt megtörténhessen:

- Az egyes szolgáltatásokhoz tudni kell tetszőleges számú forrás oldalt felvenni
- Egy új forrás oldal felvétele és felkonfigurálása, hogy mit és honnan kell letölteni nem lehet nehéz, minimális kódírással meg kell tudni valósítani
- Tudni kell kezelni az authentikációt, cookie-kat és sessionöket
- Nem szabad, hogy javascript generált, vagy ajaxos oldalak problémát okozzanak
- A letöltésnél jó lenne tudni választani, hogy a nagyobb erőforrást igénylő böngészővel való letöltés vagy egy egyszerű html file letöltéssel akarunk dolgozni
- A böngészővel való letöltés esetén célszerű, ha meg tudjuk választani, hogy Chrome, Firefox, IE vagy valami headless (phamtonjs) megoldásra van szükség

- A letöltőknek el kell tudni menteni a leszedett file-okat
- A letöltőknek érdemes cachelni, hogy ne pazarolják feleslegesen az erőforrásokat
- A letöltőknek tudniuk kell felismerni, hogy mik a már leszedett tartalmak és azokat ne járják be ismét
- Tudni kell kezelni a hibákat, szélsőséges eseteket és egy újrafuttatásnál a kimaradt elemeket letölteni
- A jellemző hibákra fel kell legyen készítve a rendszer, tudnia kell ezekre megoldást adni: 404, 500 és egyéb error-ok, DOS támadás miatti kitiltás, magasabb terhelésből adódó válaszok elmaradása
- Szükség lehet csatolmányok letöltésére is
- Legyen lehetőség adott tartalmat újra letöltetni, ha esetleg valami változás történt. Ennek indítása manuálisan történik.

3.5. Feldolgozás

A feldolgozás alapja az a html tartalom, amit az első lépésben a letöltő előállított. Ebből fogja az alkalmazás felépíteni a DOM-t, aminek a fa struktúrájában lépkedve a megfelelő csomópontig eljutva könnyen kinyerhető a szükséges adat.

A modern webes világban, ahol az oldalak strukturáltsága segíti hozzá a szolgáltatások minél jobb eladhatóságát jó alapul szolgál ahhoz, hogy a számunkra értékes adatokat könnyedén kinyerjük, hisz az adatok az oldalon minden esetben ugyan ott lesznek elérhetőek, ami azt eredményezi, hogy az adatok jól beazonosíthatóak. Ez azért fontos a feldolgozó szempontjából, mert így az adatok kinyeréséhez elégséges platformonként egy-egy feldolgozót implementálni. Természetesen, vannak kivételek és akadnak olyan oldalak, amik kevésbé strukturáltak vagy épp nem teszik elérhetővé a hirdetmény adatait html formában, helyette valamilyen letölthető dokumentumként tüntetik fel az oldalon (doc vagy épp pdf formátumban). Ezekre az esetekre egyedi megoldásokat kell adni és valamilyen szöveges illesztéssel, maszkolással oldható meg a feldolgozás.

- A feldolgozónak tudni kell parsolni a letöltött tartalmakat
- Konfigurálni kell tudni, hogy milyen adatokat kell kinyerni és azok hol és milyen módon találhatóak a tartalomban
- Minimális kódírással lehessen új feldolgozót írni
- Elhalás esetén a kimaradt feldolgozandó tartalmakat lehessen újra feldolgozni
- Hogyha hibásan történt meg egy feldolgozás, akkor lehessen megadni, hogy melyik tartalmakat kell ismételten feldolgozni
- Adott esetben lehet nem html-ben, hanem doc-ban, pdf-ben kell tudni feldolgozást végezni

- Hibahatár számítása, ami alapján meg lehet jelölni, osztályozni a hirdetményt, hogy kézi javítás szükséges

3.6. Indexelés

A felhasználók számára a releváns hirdetmények megtalálásához szűrési feltételeket kell definiálniuk. Ezek több félék lehetnek úgy, mint szöveges, szám alapú, dátum alapú esetleg valamilyen egyéni osztályozás alapú. Mivel a hirdetmények és a keresőprofilok Descartes szorzatát kell venni a találatok generálásához a keresésnél, ami egy nagyszámú iterációt jelent, így a rendszernek ezt a lehető leggyorsabban kell végrehajtani. Ehhez az egyik legjobb módszer az indexelés, ami az adatokat olyan plusz információval látja el, ami meggyorsítja a hirdetmények összerendelését a keresőprofilokkal.

- Nagy adatmennyiségekről lévén szó fontos, hogy a keresés gyors legyen
- Lehessen a megadott struktúra minden adatának keresését indexeléssel gyorsítani (ez nem jelenti azt, hogy mindent kell indexelni, csak azt amit meghatároztunk, hogy keresni is akarunk)
- Legyen fulltext search
- Legyen egyszerű szótövezés

3.7. Találatok generálása

A felhasználó keresési feltételei alapján össze kell tudni rendelni a keresőprofilokat az arra illeszkedő hirdetményekkel. Ez az összerendelés minden nap egyszer fut le és mindig az aznapi frissen megjelent hirdetményeket dolgozza fel.

- Az előre megadott keresőprofilok alapján tudjunk találatokat generálni
- Ha valami probléma van, elhal a találatok generálása, akkor lehessen újrageneráltatni ezeket a megadott feldolgozott tartalmakra (hirdetményekre) vagy időszakra
- Tudjuk meghatározni, hogy a keresőprofilon belül milyen feltételre történt a matchelés

3.8. Találat értesítő e-mailek

A kigenerált találatokról a felhasználó e-mailes értesítőben kap tájékoztatást. A levélben táblázatban megjelenik a hirdetmény részletei.

- A legenerált találatok alapján a rendszer tudja egy e-mailbe összeszedni a találatokat

- E-mail formájában az ügyfélnek lehessen kiküldeni az adott időszakra vonatkozó találatokat
- Lehessen bannereket elhelyezni a találatértesítő e-mailekben
- (Legyen lehetőség ajándék tartalom elhelyezésére is az e-mailekben)

3.9. Ütemezés

Az ütemező feladata az egyes folyamatok elindítása a megadott időpontokban. Ezek előre definiáltak és mindig ugyan akkor futnak napi bontásban, az alapján, hogy a letöltendő adatok mikor kerülnek fel a platformokra, a letöltések mikor fejeződnek be.

- Lehessen tetszőleges ütemezést könnyen felkonfigurálni, hogy melyik komponens mikor fusson (letöltés, feldolgozás, találatok generálása, találat értesítő e-mailek)

4. Alkalmazás (APP)

4.1. Regisztráció / Bejelentkezés

Az alkalmazásnak felületet kell biztosítani ahhoz, hogy a felhasználó megfelelő jogosultságot szerezhessen, azaz regisztrálhasson a rendszerbe. A regisztrációt követően be kell tudnia jelentkeznie a felhasználónév/jelszó páros megadását követően.

- A marketing oldalon (CMS) keresztül lehessen regisztrálni a rendszerbe
- A marketing oldalon és az alkalmazáson (APP) keresztül is lehessen bejelentkezni a rendszerbe
- Legyen token alapú bejelentkezés, hogy e-mailben egy linken keresztül be lehessen jelentkezni
- [admin] Lehessen más nevében bejelentkezni, hogy látni lehessen beállításait és segíteni rendbe rakni a fiókját
- Szükség van e-mail cím validálásra
- Lehessen jelszó emlékeztetőt kérni (új jelszót generálunk neki)

4.2. Beállítások

A felhasználónak legyen lehetősége a regisztrációnál megadott adatainak módosítására. Legyen lehetősége a találatértesítőkhöz több e-mail címet is beállítani.

- Lehessen az ügyfél adatokat szerkeszteni (ez leginkább az e-mail címet jelenti, de akár számlázási adatokat is módosíthat) CRM-ben
- Lehessen alternatív e-mail címeket megadni az ügyfélhez, hogy több címre is ki tudjon menni találatértesítő
- Lehet, hogy keresőprofilonként is kellene tudni megadni, hogy milyen e-mail címre menjen még ki
- Az ügyfél tudjon a rendszerben jelszót módosítani
- Lehessen megadni, hogy hogyan kell megszólítani az ügyfelet
- [admin] Lehessen elkülöníteni a belső felhasználókat, ne csak szerep szinten. Lehet például egy belső felhasználó teszttel, ahol szeretnénk látni a rendszerben, hogy mennyi belső felhasználó van, vagy mondjuk kik azok, akik nem fizetnek a szolgáltatásért.

4.3. Keresés

Ahhoz hogy a felhasználó a keresési feltételeit a lehető legpontosabban megadhassa, amire azért van szükség, hogy csak a releváns hirdetményekről kapjon értesítést, s számára érdektelen hirdetmények ne kerüljenek a találatai közé, szükség van arra, hogy a hirdetmények között tudjon keresni, azaz kipróbálhassa, hogy a megadott keresési feltételek mellett milyen hirdetményekre számíthat a találatértesítőkben.

Erre azért is van szükség, mert azonnali visszajelzést érkezne arról, hogy a megadott keresési feltételek nem adnak több eredményt az elvártnál melyek tartalmazzák az összes kívánt hirdetményt.

A keresési feltételek típusa több féle lehet, mindig az adott mezőnek megfelelő legrelevánsabb, úgy, mint dátum esetén range feltétel (tól-ig), számok esetén pontos egyezés vagy intervallum tartalmazás, vagy kisebb/nagyobb feltétel és így tovább. Szöveges keresésnél a tartalom egy része és a teljes tartalmi egyezés egyaránt megadható legyen. Az egyes feltételek egyszerre teljesülése szűri a hirdetményeket, míg egy feltételen belül bármelyik egyezése ad találatot. Például szöveges keresésnél, ha több szó is meg lett adva, akkor bármelyik szót tartalmazó hirdetmény már találatnak felel meg, míg ha meg lett adva szöveges és CPV kódos keresési feltétel is, akkor mindkettőt tartalmaznia kell az adott hirdetménynek, ahhoz hogy találatként tároljuk a felhasználóhoz.

- Kereső felület, ahol tetszőleges attribútumra keresni lehet
- A lefuttatott keresést lehessen elmenteni, hogy később újra lehessen futtatni - (Keresőprofilként lehet jelenleg menteni)
- Itt már lehessen a nem aktív hirdetmények között is keresni
- Lehessen megadni a keresés intervallumát
- Lehessen megadni, hogy aktív vagy nem aktív vagy mind között akarunk keresni

- Lehessen keresést előre definiált keresőprofil segítségével összerakni, töltődjenek be annak beállításai, amit aztán módosítani lehet
- Lehessen kiíróra, nyertesre keresni
- Lehessen forrás oldal alapján keresni
- Lehessen szavakra, szótöredékekre keresni a nagyobb szövegekben
- Attribútumokban jellemzően a teljes egyezést nézzük
- Legyen egymezős keresés, ami suggestionként az előre definiált keresőprofilokat használja, vagy ha ott nincs találat, akkor keresőszóként fulltext search
- [admin] Az admin tudja beállítani az adott plarformon, hogy mely attribútumok kereshetőek

4.4. Keresőprofil

Keresőprofil nem mást, mint a keresési feltételeket összefogó nevesített entitás. Ezeknek a száma korlátlan, viszont az aktívak száma mindig a megvásárolt előfizetési csomagtól függ.

- A keresést lehessen elmenteni, mint kereső profil (gyakorlatilag kereső profilt is úgy veszünk fel, hogy megad keresési paramétereket, tesztként lefuttatjuk és utána elmentjük, mint kereső profil)
- Lehessen keresőprofilokat inaktiválni
- Hogyha egy felhasználó előfizetése lejár, vagy csomagváltás miatt nem érhető el számára annyi keresőprofil, mint korábban, akkor automatikusan inaktiválni kell a keresőprofiljait
- Egy felhasználónak lehessen több keresőprofilja, mint amennyi aktív keresőprofilt megenged az előfizetése, csak ebben az esetben inaktív keresőprofilok vannak.
- Az aktív keresőprofilokból kell találatértesítőket generálni
- A keresőprofil attribútumai között lehessen és, vagy kapcsolatot definiálni. Továbbá lehessen kizáró feltételeket is megadni.

4.5. Keresési eredmények exportálása

- A keresésnél vagy bármi egyéb módon listában megjelenő hirdetményeket tudni kell exportálni excelbe, amibe a név, meta adatok és egy link kerül bele

4.6. Keresési eredmények kiküldése e-mailben

- Egy keresés eredményeit lehessen kiküldeni egy adott e-mail címre, mintha találatértesítő lenne
- A találatok között legyen lehetőség még kézzel válogatni és a megjelöltek kerüljenek csak kiküldésre
- [admin] Az így kiküldött levélhez lehessen megadni e-mail címet, így lehet pl egy nem regisztrált ügyfélnek mintát
- [admin] Kézzel lehessen hirdetményt hozzáadni találatértesítőhöz

4.7. Találatok megtekintése

A felhasználó az e-mailes értesítőn kívül a felületre belépve is meg tudja tekinteni találatait.

- Meg kell tudni jeleníteni a rendszerben, hogy melyik nap mentek ki e-mailek,
- Azok hány találatot tartalmaztak
- Melyik keresőprofilból mennyit
- És eggyel mélyebbre menve megnézni azt is, hogy konkrétan melyik hirdetmények voltak a találatok
- Az ügyfél csak a saját keresőprofiljai között tudja ezeket megtenni. (az admin átjelentkezéssel tudja mások találatait megjeleníteni)
- Legyen lehetőség egy elvileg kiküldött e-mail újragenerálására és újraküldésére

4.8. Hirdetmények megtekintése

A hirdetmények részleteit megtekinthetővé kell tenni a felületen, amennyiben nincs külsős hivatkozása. Ebben az esetben a forrásoldalra kell átnavigálni a felhasználót.

- A hirdetményekre kattintva el kell tudni dönteni, hogy a külső forrásra mutató linket használjuk, vagy belső oldalon jelenítjük meg (itt érdemes nem direkt linket tenni a forrás oldalra, hanem egy redirectes saját linket, amin keresztül mérni tudjuk a megtekintéseket)
- Meg kell tudni belső oldalon jeleníteni a hirdetményeket
- Hogyha a hirdetményhez vannak csatolmányok, akkor azokat tudni kell kilistázni és letölthetővé tenni

4.9. Hirdetmény letöltés

A hirdetményeket lehessen exportálni valamilyen formátumban. Ezen kívül a hirdetményeket meg lehet jelölni kedvencként, amit egy külön oldalon meg is lehet tekinteni.

- Tudni kell pdf-be exportálni a hirdetményt, hogy letölthető legyen

4.10. Kedvencek

- A hirdetményeket listában megjelenítve a meta adataikkal tudni kell a kedvencek közé rakni
- E-mailből is lehessen a kedvencek közé tenni hirdetményt
- A kedvenceket lehessen kilistáztatni
- Lehessen a kedvencekből törölni

4.11. Olvasottnak jelölés

További jelölésként a hirdetmény olvasottnak jelölhető, így az ügyfél követni tudja, melyek azok a hirdetmények, amiket korábban már megtekintett, s mik azok, amikkel még nem találkozott. Ez leginkább abban az esetben hasznos, amikor az ügyfél találatainak a száma olyan magas, hogy azok megtekintése csak részleteiben lehetséges, így az olvasottnak jelölés segíti a feldolgozás gyorsítását, hisz nem kell minden egyes hirdetményt újra megnyitnia minden megszakítás után, hisz a már megnyitottak meg lesznek jelölve.

- Egy hirdetményt megtekintve (legyen az e-mailben kattintás vagy felületen történő, akkor, ha külső linkre mutat), tudni kell eltárolni, hogy azt már megtekintette az ügyfél
- A hirdetmények listázásakor tudnunk kell jelölni, hogy mely hirdetményeket látta már a felhasználó és melyeket nem
- A hirdetmények listában történő megjelenítésekor kell gyors lehetőség arra, hogy akár tömegesen is olvasottnak jelöljön hirdetményeket az ügyfél
- Egy hirdetményt lehessen nem olvasottnak jelölni (hasonló funkcionalitás, mint levelezőrendszerekben e-maileknél)

4.12. Hirdetmények követése

Minden egyes közbeszerzési kiírás több hirdetményből tevődik össze az aktuális státusza alapján. Ezeknek a követhetőségét kell biztosítani, azaz a felhasználót értesíteni a kiíráshoz tartozó újabb hirdetmény megjelenéséről.

- Egy hirdetmény kiírás kapcsolódhat korábbi hirdetményhez, mert újabb adatközlésről lehet szó. Lehet ez nyertes hirdetése, vagy módosítás, stb. A felhasználónak jó lenne tudni megjelölni, hogy egy hirdetményt követni szeretne.
- A követett hirdetmények kiemelten jelennek meg az ügyfél számára mind a felületen, mind az e-mailek között

4.13. Kiírók és nyertesek megjelenítése

A közbeszerzések kiírói és a közbeszerzést elnyerő szervezeteket kereshetővé kell tenni. Ehhez kapcsolódóan pedig le lehessen kérni, hogy a mind a kiíró, mind a nyertes milyen további hirdetményekben szerepelt korábban.

- A rendszernek tudni kell keresni nem csak a hirdetményekben, de a kiírókban és nyertesekben is
- Lehessen kilistáztatni a kiírókat és nyerteseket
- Lehessen megnézni, hogy egy kiíróhoz vagy nyerteshez milyen hirdetmények tartoznak
- Jelenjen meg a listás nézetben, hogy melyik kiíróhoz és nyerteshez mennyi hirdetmény tartozik
- Lehessen rendezni a hirdetmények száma alapján

4.14. Kiírók és nyertesek követése

Hasonlóan a hirdetményekhez a kiírókat és a nyerteseket is lehessen követni, azaz olyan hirdetmény megjelenésénél is kapjon értesítést a felhasználó, ami a megjelölt kiíróhoz vagy nyerteshez kötődik.

- Lehessen kiírókat és nyerteseket is követésre jelölni
- Amennyiben olyan hirdetmény kerül a rendszerbe, ami kiíróként vagy nyertesként az adott hirdetményhez tartozik, úgy arról kapjon találatértesítést az ügyfél és az ilyen legyen kiemelve benne

4.15. Előfizetések menedzselése

Ahogy arról már korábban is szó esett, a hirdetmények értesüléséhez aktív előfizetéssel kell rendelkeznie a felhasználónak, amihez csomagot kell vásárolnia. Ezt a felületen keresztül teheti meg akkor, ha éppen nincs aktív előfizetése, illetve a rendszer automatikusan meghosszabbítja az ÁSZF-ben foglaltak szerint. Az előfizetés alapján aktiválhatók a keresőprofilok. Admin oldalról a csomagok kialakítására kell felületet biztosítani, illetve azok módosítására, törlésére.

- Meg kell tudni jeleníteni, hogy milyen előfizetése van az illetőnek és annak meta adatait, például mikor jár le, hány keresőprofilja lehet
- Előfizetésnek megfelelő korlátozásoknak tudni kell működjön a rendszerbe (pl.: meddig érvényes, mennyi kereső profilja lehet, maximum mennyi találatot küldhetünk ki neki, exportálhat-e excelbe, maximum mennyi sort exportálhat) . CRM
- Az ügyfélnek tudnia kell igényt leadni egy magasabb előfizetésre
- [admin] Tudni kell előfizetési csomagokat definiálni, amiben megadható, hogy mennyi keresőprofil vehető fel és milyen hosszú ideig tart az előfizetés

4.16. Dashboard

Ennek a felületnek a feladata a felhasználó informálása az aktuális státuszáról a rendszerben, azaz előfizetési adatairól, keresőprofiljairól, találatairól egy összefoglaló nézet. A nézetről tovább lehet navigálni a feltüntetett adatok részletes nézeteire.

- Bejelentkezés után legyen egy dashboard felület, ahol az ügyfél látja a fontosabb információkat
- Jelenjen meg, hogy mikor jár le a hirdetése
- Mennyi keresőprofilja van
- A keresőprofilok mennyi új olvasatlan hirdetményt tartalmaznak
- Grafikonos megjelenítés, hogy melyik napra hány találat esett
- Kedvencek megjelenítése
- Követett kiírók és nyertesek
- Info buborék, belső tájékoztatásra

4.16.1. [admin] Előre definiált keresőprofilok

Ahhoz hogy a felhasználókat segítse a rendszer a megfelelő keresőprofilok létrehozásában, előre definiált profilokat definiál, amiket akár sajátjaként elmenthet a felhasználó, s felhasználhat a kereséseihez. Ezek a korábbi tapasztalatok alapján lettek összeállítva minden féle területekre specializálva, hogy a lehető legrelevánsabb találatokat adják. Ezeket az adminisztrátor egy külön felületen tudja feltölteni és módosítani.

- Legyen lehetőség előre definiálni keresőprofilokat nevekkel
- Az előre definiált keresőprofilokat tudják az előfizetők felhasználni, mint keresőprofil
- Legyenek publikus keresőprofilok, amik találatainak meta adatai listában megjeleníthetőek a nem regisztrált ügyfeleknek – CMS

4.16.2. [admin] Offline hirdetmények feltöltése

Azoknak az eseteknek a kezelésére, amikor a hirdetmények nem jelennek meg online felületen keresztül (helyette nyomtatott formában vagy egyéb módon) felületet kell biztosítani az adminisztrátorok számára, hogy ők maguk is tölthessenek fel hirdetményeket. Ezeket ugyan úgy kell kezelnie a rendszernek, mintha az automatizált módon lettek volna feltöltve és ugyan azokat az adattagokat lehet felvenni hozzájuk, mint az automatizált megfelelőjüknél.

- Legyen lehetőség a rendszerbe manuális betölteni hirdetményeket
- Lehessen az így feltöltött hirdetményekhez csatolmányokat feltölteni
- A ilyen módon feltöltött hirdetmények ugyan úgy kezelődjenek, mint a letöltők által a rendszerbe kerülők. Azaz legyenek indexelve, tudjanak találatértesítő bekerülni és lehessen rájuk keresni

4.16.3. [admin] Hirdetmények szerkesztése

Mivel a hirdetmények közzétételénél előfordulhat, hogy azok hibásan kerülnek fel, szükség van arra, hogy azokat lehessen manuálisan javítani, módosítani. Ehhez az adminisztrátoroknak külön elérhetőséget kell biztosítani, ahol kiválaszthatják a szerkeszteni kívánt hirdetményt, majd a módosításokat eltárolhatják.

- Lehessen a rendszerben lévő automatikusan és kézzel betöltött hirdetményeket szerkeszteni (többször előfordul, hogy a parsolás nem tökéletes és javítani szükséges)
- Mentés után az indexelést ha szükséges újra meg kell csinálni

4.16.4. [admin] Más user nevében bejelentkezés

- Az admin felhasználóknak legyen lehetőségük más ügyfél fiókjába bejelentkezni, hogy pontosan azt lássák, amit az ügyfél
- Így lehetőség nyíljon arra, hogy telefonos segítségnyújtással keresőprofilok és egyéb beállítások legyenek végrehajthatók

4.16.5. [admin] Admin felület

Adminisztrátorok számára a dashboardhoz hasonlóan ki kell alakítani egy olyan nézetet, ahol a felhasználókról kaphatnak információt és amin keresztül a rendszer beállításait és felhasználóit módosíthatják, ide értve a tartalmakat, előfizetéseket, előfizetési csomagokat, felhasználói adatokat, leveleket.

- Legyen egy csak adminok számára elérhető felület
- Lehessen kilistáztatni a felhasználókat
- Logolt eseményeket lehessen nézni
- Fordítással kapcsolatos feladatokat elvégezni
- Előfizetési csomagokat definiálni - CRM, CMS
- Előfizetéseket menedzselni - CRM
- Előre definiált keresőprofilokat menedzselni
- Statisztikai adatokat elemezni
- Levél sablonok szerkesztése (pl.: találatértesítő)
- CPV kódok menedzselése
- NUTS kódok menedzselése

4.16.6. [admin] Statisztika

Elemzésekhez készíteni kell egy statisztika oldalt, ahol a rendszerről és azt használókról kapunk képet.

- Lehessen az alkalmazással kapcsolatos statisztikai adatokat megjeleníteni
- Hogy alakul a regisztrált felhasználók száma
- Hogy alakul az aktív előfizetések száma
- Hogy aránylanak egymáshoz a különböző előfizetési csomagok

- Mikor mennyi letöltés történt
- Mikor mennyi hiba volt a crawlerben (letöltés, feldolgozás, keresés, stb)
- Mikor mennyi találatot generált a rendszer
- Mikor mennyi e-mail került kiküldésre
- Aktív és passzív keresőprofilok aránya
- Aktív és passzív felhasználók aránya
- Felhasználói hibaüzenetek gyakorisága
- Egyéb

4.16.7. [admin] Levél sablonok szerkesztése

A rendszer a felhasználókat e-mailen keresztül értesíti a találatokról, az előfizetésük aktuális státuszáról vagy épp azok változásairól. Ezeknek a tartalmát szerkeszthetővé kell tenni egy külön oldalon.

- Lehessen levél sablonokat szerkeszteni (pl: hogyan nézzen ki egy találatértesítő levél)
- Lehessen a levél sablonba placeholdereket elhelyezni

4.17. Többnyelvűsítés (i18n)

Mivel a rendszer nemzetközi platformra készül, így biztosítani kell az oldalak több nyelven való elérését. Itt arra kell törekedni, hogy egy új országban történő megjelenés a fordítás szempontjából a lehető leggyorsabb legyen, azaz egyszerűen lehessen az oldalt új nyelven elérhetővé tenni.

- Legyen lehetőség tetszőleges számú nyelven megjeleníteni az oldal statikus tartalmait (menü, előre megírt szövegek, hibaüzenetek, stb.)

4.18. API

- Legyen lehetőség API-n keresztül kiajánlani funkcionalitásokat partnereknek

5. Marketing oldal (CMS)

CMS rendszer használata főleg a bevonni kívánt szakemberek redukálásáért lett kiválasztva, azaz a tartalmak létrehozása vagy módosítása ne igényeljen fejlesztői erőforrást. Fontos szempont volt továbbra, hogy a módosítások ne okozzanak hibát a rendszerben, azaz a tartalom szerkesztője korlátozva legyen és más műveletet ne hajthasson végre, tehát kódolnia ne kelljen. CMS rendszer továbbá modularizált, azaz kisebb építőkockákkal egyszerűen bővíthetőek, így a későbbiekben felmerülő felhasználói igények gyorsan kielégíthetővé válnak.

5.1. Alap CMS funkcionalitások

- Lehessen page-ket menedzselni
- A módosítások verziózva legyenek, így vissza lehet azokat állítani
- Lehessen widgetek-et kirakni tetszőleges container-be
- Lehessen többszintű menüt kezelni
- Legyen felkészítve a többnyelvűségre (i18n)
- Lehessen híreket, blog-ot kezelni benne
- Legyen benne file és média (képek, videó) kezelés
- Támogassa a SEO-t, lehessen az oldalaknak szép url-eket adni, meta tag-eket beállítani
- Egy átállás esetén lehessen megadni, hogy a korábbi linkek az új oldalra tudjanak mutatni (redirect)
- Google Analytics és Adwords követőkódok elhelyezésére legyen lehetőség

5.2. Webshop

Az előfizetések megvásárlásához webshop kerülne kialakításra, amiben termékenként megvásárolhatóak a különböző csomagok. A csomagok a webshopban létrehozhatóak, módosíthatóak. A megvásárolt csomagok rendelések formájában listázhatóak. Rendelésekhez státuszok definiáltak, melyekkel lekövethető, hogy a felhasználó a csomag ellenértékét befizette e már. Azért is esett a választás egy ilyen megvalósításra, mert az ügyfelek nagy része találkozott már webshoppal, így könnyen el fognak igazodni a csomagok megvásárlásán. Az ügyfelek megtartásához vagy épp megnyeréséhez a megvásárolható termékekhez kedvezményeket lehessen definiálni, illetve kuponokat lehessen beváltani a rendeléseknél.

- Lehessen szolgáltatás alapú termékeket felvenni, itt a terméknél nincs kiszállítás, online szolgáltatásra fizet elő
- Lehessen akciózni (leárazás, kedvezmény) a termékeket

- Lehessen kuponokat kiosztani, amivel egyedi kedvezmények adhatóak és nyomon követhetőek az egyes marketing kampányok
- Mindenképpen szükséges ismerni, hogy melyik ügyfél milyen csatornán, marketing kampányon keresztül jött (érdemes erre google okos linkeket és kuponokat, affiliate kódokat használni)
- Legyen PayPalos fizetés támogatás
- Lehessen menedzselni a megrendeléseket, és azok státuszait
- Lehessen értesítést küldeni az ügyfélnek a megrendelés státusz változásokról
- Legyen lehetőség proforma és rendes számla kiállításra (e-számlázó integráció)
- Legyen egy API, aminek segítségével az Alkalmazással (APP) és CRM-mel is össze lehet kötni
- Legyen képes ügyfél adatok bekérésére és azok testreszabására
- Statisztika a webshophoz
- Legyen lehetőség ÁSZF megadására és annak elfogadtatására

5.3. Közbeszerzés figyelő modul

A modul szerepe keresési feltételek alapján az ügyfelek számára csalogató hirdetmények megjelenítése az oldalon. Ezek megjelenítése történhet listázva, területi vagy akár időrendi viszonylatban.

- Lehessen megjeleníteni NUTS kódok (terület) alapján, hogy hol mennyi aktív hirdetmény van
- Lehessen a jövőben térképen megjeleníteni a hirdetmények számát (pl: Google Maps)
- Lehessen előre definiált keresőprofilok mentén megjeleníteni, hogy melyik profil hány aktív hirdetményt tartalmaz
- Lehessen kilistáztatni a NUTS kódok és előre definiált keresőprofilok alapján a hirdetményeket, amiknek csak meta adatai jelenítődnek meg
- Lehessen naptár nézetben megjeleníteni a hirdetményeket
- Partneroldalakon elhelyezhető widgetek beágyazó kóddal

6. Ügyfél adatbázis (CRM)

Az ügyfelek folyamatainak kezelésére alkalmas rendszer. Segítségével meglévő és új ügyfelek adatait képes tárolni, az ügyfelek felé irányuló folyamatokat támogatja. Ezek azok a

folyamatok, amik az együttműködéshez köthetőek, úgymint a megkeresések, a szolgáltatás értékesítése, meglévő előfizetések megújítása.

6.1. Ügyfelek menedzselése

- Lehessen a rendszerbe ügyfelek felvenni és azok adatait kezelni
- Lehessen egyedi mezőket definiálni az ügyfelekhez
- Lehessen az ügyfelekhez státuszt rendelni, annak history-ját megtekinteni
- Lehessen beállítani az ügyfélhez, hogy kihez tartozik, melyik ügyfélszolgálatoshoz
- Lehessen olyan kontaktokat felvenni a rendszerbe, akik még nem ügyfelek, de potenciális ügyfelek és EDM-ezni, hideg hívni szeretnénk őket
- Lehessen excelből, csv-ből ügyfél adatbázist feltölteni

6.2. Szolgáltatások managelés

- Az alkalmazásban legyen lehetőség több szolgáltatást is megadni
- Alkalmazásonként több termék (előfizetés) is legyen definiálható és paraméterezhető
- Az ügyfelekhez lehessen több szolgáltatást / terméket hozzárendelni
- Ügyfél és szolgáltatás alapon is lehessen lekérdezés csinálni, hogy látható legyen, melyik szolgáltatásokhoz milyen ügyfelek tartoznak és melyik ügyfélhez milyen szolgáltatások

6.3. Termékek / Előfizetések menedzselése

- A rendszerben lehessen termékeket és előfizetéseket is definiálni
- Az ügyfelekhez ne csak terméket / előfizetést lehessen rendelni, lehessen azt is tudni, hogy ki mikor mire fizetett elő és hogy meddig tart a szerződése
- Lehessen a rendszerben egyedi előfizetéseket beállítani, egy előfizetés határidejét meghosszabbítani menet közben, vagy ingyenes (teszt) előfizetést adni
- Legyen lehetőség proforma és rendes számla kiállításra (e-számlázó integráció)

6.4. Automatikus előfizetés hosszabbodás

A regisztráció foganatosításánál az ügyfél elfogadja, hogy a rendszer automatikusan létrehozza az ügyfélnek a korábbi előfizetése alapján, annak meghosszabbításaként az új előfizetését, mely egyben arra is kötelezi, hogy a létrehozott előfizetéssel kötelezettséget is vállal annak árának megtérítésére. Ezt az alkalmazás napi szinten kell ellenőrizni minden és olyan felhasználónak automatikusan legenerálni a webshopba a rendelést, akiknek az előfizetése lejárt. Akkor is, ha teszt előfizetéssel rendelkezik, vagy le fog járni hamarosan, ha normál előfizetése van. Az ügyfélnek bizonyos esetekben lesz lehetősége a létrehozott előfizetés lemondására. Minden egyes létrehozott előfizetésről az ügyfél e-mailes értesítést kap, illetve a felületre belépve is tájékoztatást kap a meghosszabbítás tényéről és annak vonzatáról. A rendszer továbbá felszólító e-maileket küld bizonyos időközönként minden ki nem egyenlített tartozásról. A hosszabbítást végző APP minden további rendszerben rögzíti (szinkronizálja) a létrehozott új előfizetést.

- A rendszer legyen képes automatikusan meghosszabbítani egy előfizetést, ha az a megadott időpont előtt nem került lemondásra

6.5. Taskok menedzselése (call center funkciók)

- Lehessen a rendszerben az ügyfélszolgálatnak, call centernek feladatokat felvenni, azokat priorizálni és határidővel ellátni
- Felhívandó kontaktokból is lehessen feladatok csinálni
- Generálódhasson automatikusan feladatok (taskok), ha bizonyos feltételek teljesülnek, pl valakinek lejár az előfizetése 1 hónap múlva és emiatt fel kell hívni
- A taskokat lehessen átallokálni másokhoz, azt lehessen nyomon követni, hogy ki mikor mit csinált az adott feladattal
- Lehessen a taskok alapján felhasználói statisztikát készíteni, pl egy hónapban ki mennyi feladatot és milyen típusút kezelt / oldott meg

6.6. Hibabejelentések kezelése

- Legyen lehetőség ügyfélhez berögzíteni hibabejelentéseket
- Lehessen többszintű supportot megvalósítani a rendszerben (Support Level 1-2-3-4)
- A rendszer tudjon e-mailből hibabejelentést generálni

6.7. Levél küldés

- Lehessen a rendszerben sablon leveleket definiálni

- Lehessen eseményekhez leveleket rendelni, amiket a rendszer képes automatikusan kiküldeni az ügyfélnek
- Hibabejelentések állapotváltozásáról lehessen a rendszeren keresztül értesíteni az ügyfeleket

6.8. Figyelmeztetések

- A rendszer legyen képes figyelmeztetéseket generálni valami esemény bekövetkezésekor. Pl.: túl sok a lekezeletlen feladat valakinél, az ügyfélnek mindjárt lejár az előfizetése
- A figyelmeztetésekből tudjanak feladatok (taskok) generálódni

6.9. Statisztika

- Lehessen statisztikákat lekérni a rendszerből szolgáltatáshoz, termékhez, ügyfélhez, felhasználóhoz (ügyfélszolgálatos)

6.10. Export / Import

- Legyen lehetőség a rendszerből adatokat exportálni és importálni, elsődlegesen excel formátumban

6.11. EDM (integráció)

- A rendszer vagy legyen képes EDM kampányok kiküldésére
- Vagy lehessen az ügyfeleket az adataikkal együtt egy másik EDM alkalmazásba szinkronizálni
- A szinkronizáláshoz lehessen egyedi feltételeket megadni, hogy mikor szabad valamit szinkronizálni, illetve azt is, hogy milyen csoportba kell azt megtenni
- Lehessen az ügyfeleknek EDM kampányokat küldeni

6.12. Webshop (integráció)

- A rendszer tudja megfeleltetni a Webshopban található csomagokat a CRM-ben találhatóakkal, ezeket tudja szinkronban tartani
- A webshopban történt vásárlás (előfizetés) adati jussanak el a CRM-be

6.13. API

- A CRM rendelkezzen olyan API-val, amin keresztül egyedi szinkronizációs modulok valósíthatóak meg. Ilyen lehet pl a KHF alkalmazás (APP) és a CRM közötti adatok átadása, függvények meghívása.

6.14. Többnyelvűség (i18n)

- Lehessen a CRM felületének nyelvezetét testre szabni, legyen támogatás különböző nyelvekre, lévén, hogy több országból más-más nyelven használnák a kollégák
- Legrosszabb esetben az angol nyelv kell az alap legyen

6.15. Jogosultság, láthatóság

- A rendszernek tudnia kell felhasználói csoportokhoz jogosultságokat rendelni, hogy milyen funkciót érhet el a felhasználó és milyet nem. Például az ügyfelek törlése nem biztos, hogy mindenkinek engedélyezett
- A rendszernek szolgáltatásokhoz, termékekhez és ügyfelekhez is támogatnia kell a láthatóság biztosítását. Lehetnek olyan felhasználók, akik csak bizonyos szolgáltatásokkal kapcsolatos adatokat láthatnak

6.16. Affiliate rendszer kialakítása

- Az ügyfelek tudják ajánlani a szolgáltatást másoknak
- Az ajánló kapjon ingyen előfizetés hosszabbítást
- Az ajánlott kapjon kedvezményt a szolgáltatásból, vagy meghosszabbított tesztidőszakot
- Le kell tudni követni, hogy ki kit hozott a rendszerbe

7. EDM

Marketinges tevékenység egyik kiemelten fontos eszköze a felhasználók e-mailes megkeresése előfizetési ajánlatokkal. Ehhez arra van szükség, hogy tömegesen lehessen kiküldeni az e-maileket és azokat célzottan valamilyen kritériumi besorolás alapján. Ezeket lehessen előre ütemezetten, de akár manuálisan is kiküldeni bármikor.

7.1. Felhasználói csoportok

- Lehessen felhasználói csoportokat létrehozni
- Azonos felhasználói csoportba tartozó embereknek azonos (hasonló) e-maileket küldünk ki
- A felhasználók egységes adatstruktúrával rendelkeznek csoporton belül

7.2. Felhasználói adatok

- Lehessen csoporton belül tetszőleges attribútumokat definiálni a felhasználókhoz
- Lehessen regisztrációs formot generáltatni a megadott attribútumok alapján
- Lehessen kézzel is beregisztrálni és szerkeszteni felhasználókat

7.3. Szűrők

- A felhasználóhoz definiált attribútumokra lehessen szűrni
- Lehessen szűrőket létrehozni, ahol a megadott szűrők mentén szabályozható, hogy kinek, mikor milyen levél menjen ki

7.4. Levél sablonok

- Lehessen WYSIWYG editor segítségével levél sablonokat készíteni
- Legyenek a rendszerbe előre összeállított designos levél sablonok

7.4.1. Ütemezés, levél küldés

- Lehessen manuálisan megadott pillanatban levelet küldeni
- Lehessen előre megadott dátumon levelet küldeni
- Lehessen egy esemény bekövetkezésekor levelet küldeni (pl.: regisztrációt követő 5. napon, valamelyik dátum attribútum elérte előtt 1 hónappal, stb)

7.5. API

- Rendelkezzen olyan API-val, aminek segítségével importálni / exportálni és szinkronizálni tudunk adatokat

- Esetlegesen a levélküldést is lehessen vezérelni az API-n keresztül

7.6. Analitika

- A szolgáltatás minden pontját tudni kell monitorozni, analizálni, hogy ezek alapján üzleti döntéseket lehessen hozni
- A CMS esetén a Google Analytics + a webshop saját statisztikai adataira van szükség
- Az Alkalmazás (APP) esetén szintén Google Analytics és a rendszer saját maga által szolgáltatott statisztikai adatokra van szükség
- Míg a CRM esetén a CRM által biztosított statisztikákat kell elemezni tudni

7.7. Integráció

Mivel az alrendszereknek kommunikálnak egymással, aminek alapja minden esetben a felhasználó és a hozzá tartozó entitások (úgymint számlázási adatok, keresőprofilok, előfizetések, találatok), így ezeknek mindig ugyan olyan állapotba kell tartani a felhasználóról tárolt információkat, azaz bármelyik alrendszerbe változás történik a felhasználóra vonatkozóan, annak a változásnak a többi alrendszerben is meg kell jelennie (amennyiben az alrendszer érintett a változást illetően).

Ez azt jelenti, hogy a változásokat át kell tudni szinkronizálni az alrendszerekbe automatikusan, így az alrendszerek implementációjánál és a megvásárolt alrendszereknél egyaránt szempont volt, hogy azok integrálhatóak legyenek, tehát képes legyenek értesítést küldeni a változásról, hogy az azt érintő rendszerek befrissítsék a felhasználó adatait. Ilyen például az előfizetés, ami minden alrendszerre kihat, emiatt szinkronizálni kell az adatokat az APP-ból kiindulva a Webshopon és a CRM rendszeren keresztül az EDM rendszerrel bezárólag mindenhol. Mivel a szinkronizáció során bármilyen hiba léphet fel, erről értesíteni kell a rendszernek az adminisztrátorokat, akik pótolhatják a kimaradt szinkronizációs lépéseket.

- A CMS webshopnak a CRM-mel kell tudnia szinkronizálni megrendeléseket és ügyfél adatokat
- Az Alkalmazásnak szintén szüksége van az ügyfél néhány adatára (név, e-mail, stb) és arra, hogy milyen csomagra fizetett elő, meddig tart az előfizetése. Ezt az információt vagy a CRM-ből, vagy a CMS webshopból kell kapnia, bár valószínűleg a CRM-ből praktikusabb, mert ott lehet menedzselni az ügyfél adatait és előfizetését
- A CRM-ből tudni kell az EDM rendszerbe ügyfél adatokat eljuttatni, hogy leveleket lehessen nekik küldeni

7.8. Monitorozás

Mivel a rendszer sok komponensből épül fel és azok külön konténerbe futnak, így ezek monitorozása elengedhetetlen a folyamatos működés szempontjából. Egyrészről szükség van arra, hogy a rendszer jelezze, hogy ha bármelyik modul leállt, a lehető legrövidebb időn belül, hogy a supportért felelős kolléga mielőbb el tudja indítani az adott modult, hogy ismételten képes legyen kiszolgálni az ügyfelek kéréseit. Másrészről információval rendelkezzünk a rendszer erőforrás fogyasztásáról, amiből eldönthető, hogy van e szükség további erőforrás bevonására.

- A rendszereket lehessen könnyen monitorozni
- A működés szempontjából releváns információkat grafikonon lehessen vizualizálni (pl.: letöltések száma, találatok száma, kiküldött levelek száma, bejelentkezések száma, CPU, memória egyéb HW elemek terheltsége)
- A kritikusnak meghatározott eseményekről lehessen értesítéseket küldeni (e-mail, sms, telefon)

8. Elosztott működés

- A rendszer úgy legyen megtervezve, hogy fel legyen készítve az elosztott működésre
- Nem csak komponens szinten lehessen szerverekre szétválasztani, hanem a kritikusabb komponensek (crawler, parser, indexelő) által okozott terheléseket akár több szerver között el lehessen osztani
- Szükség esetén tudjunk átállni felhő alapú szolgáltatásra, hogy a szükséges erőforrásokat könnyen lehessen bővíteni

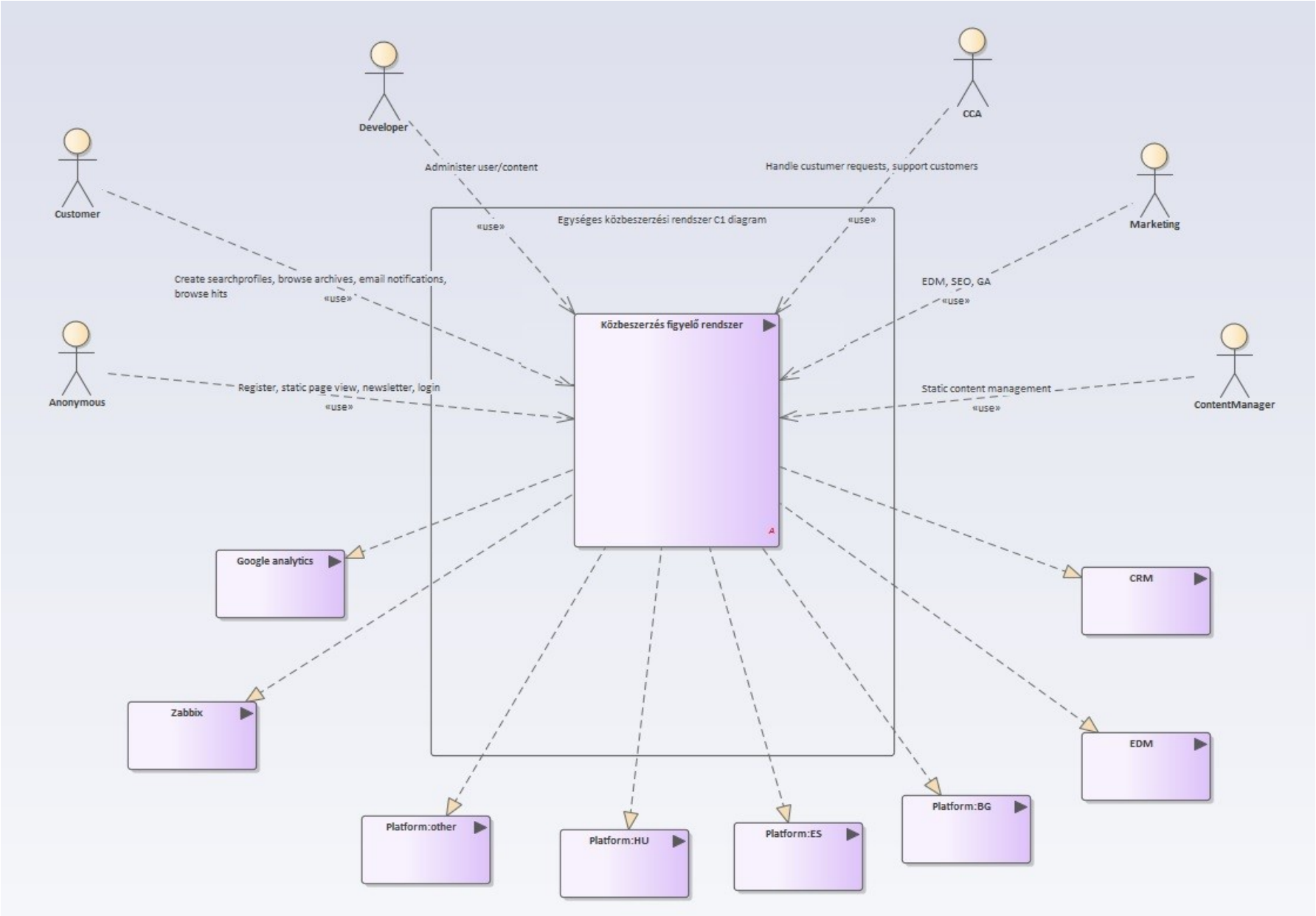
9. Nem funkcionális követelmények

- A rendszer legyen képes 100.000 felhasználót kezelni
- A rendszer legyen képes 1.000 forrás oldalt kezelni
- A rendszer legyen képes egy időben 1.000 bejelentkezett felhasználót kezelni, azok keresési igényeit kiszolgálni
- A marketing (CMS) és alkalmazás (APP) oldalak vállalható grafikával jelenjenek meg
- Ugyan ezen oldalaknak a betöltési idejük és használat közben adott válaszidejük tizedmásodpercekben legyen mérhető, ne legyen lassúság élmény. Ha valami művelet sokáig tart, akkor pedig visszajelzést kell adni a felhasználónak

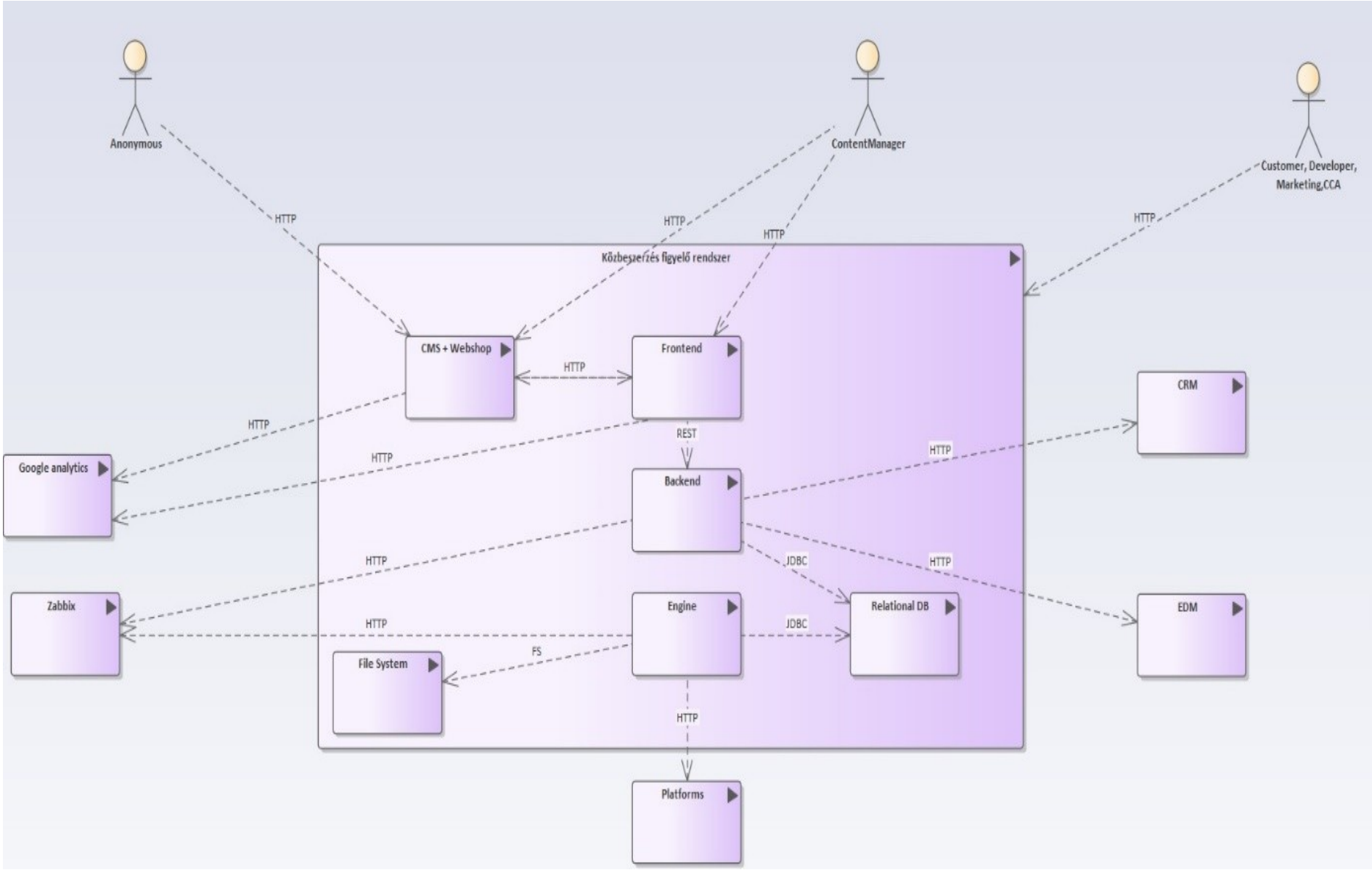
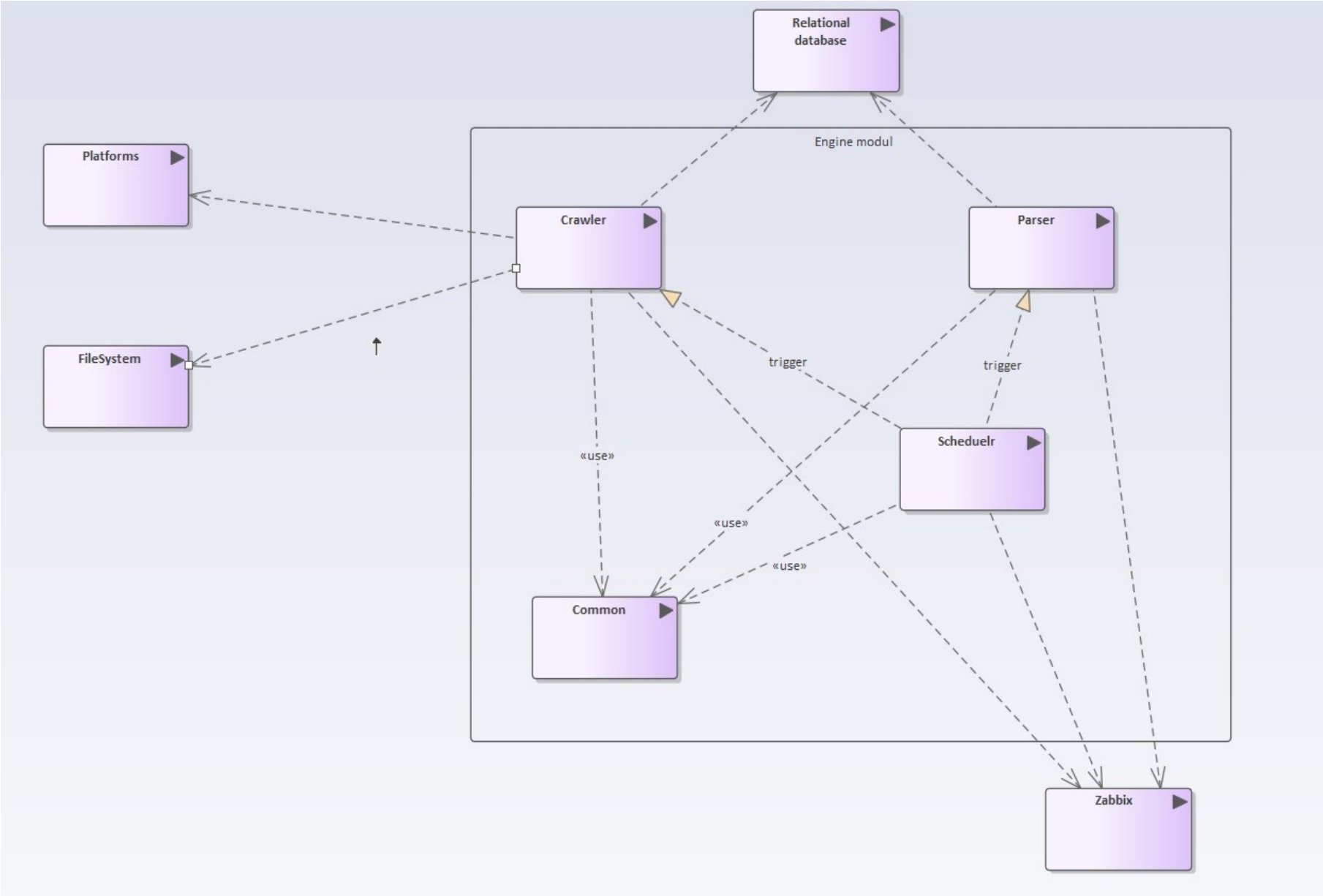
- A felületek legyenek validátorokkal ellátva
- Az ügyfél a hibákról kapjon értelmezhető hibaüzenetet (ezeket a hibaüzeneteket érdemes logolni)
- Az egyese platformok azonos kódbázison alapuljanak
- A hibák és warningok nagy részét a monitorozó rendszerek segítségével automatikusan vegyük észre
- Böngészőfüggetlenség (IE9+ legyen támogatott)
- A rendszer legyen kellően letesztelve, rendelkezzen unit tesztekkel, integrációs tesztekkel és stressz tesztekkel
- Könnyen és gyorsan lehessen deployolni, szükség esetén könnyű legyen a visszaállás
- Az éles szerver mellett legyen egy fejlesztői (Dev) és egy felhasználói (UAT) teszt szerver
- A crawlerezéssel kapcsolatos funkciókat lehessen időben elosztani, hogy ne pillanatnyi nagy szerverterhelések legyenek
- Az oldalak legtöbbje rendelkezzen nyomtatási nézettel, hogy például egy keresés esetén a listát és a keresési beállításokat lehessen látni
- Ne lehessen a rendszerből az összes hirdetményt könnyen kinyerni egy jó beállítással. Limitekkel kell dolgozni az e-mail küldésnél, listák megjelenítésénél, vagy export funkcióknál.
- Redundancia megteremtése

MELLÉKLETEK

I melléklet: C1 diagram



II. melléklet C2 diagram:



III. melléklet: Engine alrendszer kibontása: