

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

NEUMANN JÁNOS
INFORMATIKAI KAR

SZAKDOLGOZAT

**OE-NIK
2020**

Hallgató neve:
Hallgató törzskönyvi száma:

Szabó Norbert Attila
T/005261/FI12904/N

Óbudai Egyetem
Neumann János Informatikai Kar
Kiberfizikai Rendszerek Intézet

SZAKDOLGOZAT FELADATLAP

Hallgató neve:	Szabó Norbert Attila
Törzskönyvi száma:	T/005261/FI12904/N
Neptun kódja:	I7XM0T

A dolgozat címe:

Menedzseri és seniori tervező és lekérdező rendszerek
Senior planning and query systems

Intézményi konzulens:	Légrádi Gábor
Külső konzulens:	

Beadási határidő:	2021. december 15.
-------------------	--------------------

A záróvizsga tárgyai:	Számítógép architektúrák Big Data és üzleti intelligencia specializáció
-----------------------	---

A feladat

Tervezzon, és valósítson meg egy olyan számítógépes rendszert, amely segíti egy cégnél a menedzser, valamint a senior munkatársak tevékenységének tervezését és monitorozását, valamint elemzését. Tervezze meg a rendszer adatait kezelő, ahhoz illeszkedő adatbázist, majd valósítsa is meg. Tervezze meg az adatbázishoz kapcsolódó backend felépítését, majd készítse el. Készítsen teszteseteket a backendhez. Készítsen a céges működéssel kapcsolatos adatokat kezelő adattárházat. Az adattárházat felhasználva készítsen egy reporting rendszert. Tesztelje a reportokat abból a célból, hogy megfelelő értéket hoznak-e. Készítse el a fejlesztői dokumentációkat. Készítse el a felhasználói dokumentációkat.

A dolgozatnak tartalmaznia kell:

- az elkészítendő rendszer tervét,
- az adatbázis struktúrát,
- a backend rendszer tervét,
- a tesztelési tervet,
- tesztelési jegyzőkönyvet,
- a megvalósítandó adattárház tervét,
- az adattárház megvalósításának dokumentációját,
- egy reporting rendszert tervét,
- a reporting rendszer megvalósításának dokumentációját,
- fejlesztői dokumentációkat,
- felhasználói dokumentációkat.



Fleiner Rita
.....
Dr. Fleiner Rita
intézetigazgató

A szakdolgozat elévülésének határideje: **2023. december 15.**
(OE TVSz 55.§ szerint)

A dolgozatot beadásra alkalmasnak tartom:

.....
külső konzulens

Hyndor Gábor
.....
intézményi konzulens



HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a szakdolgozat / diplomamunka saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült szakdolgozatomban / diplomamunkámban található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja.

Budapest, 20 21. 12. 15

hallgató aláírása



KONZULTÁCIÓS NAPLÓ

Hallgató neve: SZABÓ NORBERT ATILA Neptun kód: 17X40T Tagozat: EST1

Telefon: Levelezési cím (pl: lakcím):

Szakdolgozat / Diplomamunka¹ címe magyarul:

KEZELÉSI ÉS SENIORI TERVEZŐ ÉS LEKÉRDEZŐ RENDSZEREK

Szakdolgozat / Diplomamunka² címe angolul:

SENIOR PLANNING AND QUERY SYSTEMS

Intézményi konzulens:

Külső konzulens:

LEGÁDI GÁBOR

Kérjük, hogy az adatokat nyomtatott nagybetűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2020.09.08	RENDSZER ÉS SZAKDOLGOZAT ¹ ÁTTEKINTÉSE	Legádi Gábor
2.	2020.10.01	DOKUMENTÁCIÓ ÁTTEKINTÉSE	Legádi Gábor
3.	2020.11.03	FUNKCIÓK ÉS DIAGRAMOK MEGVÉTELÉSE	Legádi Gábor
4.	2020.11.18	VÉGSO JAVÍTANDÓ DOLGOK ÁTTEKINTÉSE	Legádi Gábor

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat I. / Szakdolgozat II. (BSc) vagy Diplomamunka 1 / Diplomamunka 2 / Diplomamunka 3 / Diplomamunka 4³ tantárgy követelményét teljesítette, beszámolóra / védésre⁴ bocsátható.

Budapest, 2021. 12. 15.

Legádi Gábor
intézményi konzulens

¹ Megfelelő aláhúzendő!

² Megfelelő aláhúzendő!

³ Megfelelő aláhúzendő!

⁴ Megfelelő aláhúzendő!



KONZULTÁCIÓS NAPLÓ

Hallgató neve: SZABÓ LORBERT ATTILA Neptun kód: 17X40T Tagozat: ESTI

Telefon: [REDACTED] Levelezési cím (pl: lakcím): [REDACTED]

Szakdolgozat / Diplomamunka¹ címe magyarul:

KEVERÉSI ÉS SENIORI TERVEZŐ ÉS LEKÉRDEZŐ RENDSZER

Szakdolgozat / Diplomamunka² címe angolul:

SENIOR PLANNING AND QUERY SYSTEMS

Intézményi konzulens:

LEGRAFI GÁBOR

Külső konzulens:

Kérjük, hogy az adatokat nyomtatott nagybetűkkel írja!

Alk.	Dátum	Tartalom	Aláírás
1.	2021.09.13.	FEJLESZÉS SZÜKSÉGES KEZDŐ INFOK ATBESZÉLÉSE	Legrafi Gábor
2.	2021.10.11.	TOVÁBBI MEGVALÓSÍTÁSHOZ SZÜKSÉGES KONZULTÁCIÓ	Legrafi Gábor
3.	2021.11.15.	RIPOORTOK KÉSZÍTÉSÉVEL KAPCSOLATOS KONZULTÁCIÓ	Legrafi Gábor
4.	2021.12.08.	VÉGSO ÁTTEKINTÉS ÉS VÉGSO VÁLTOZTATÁSOK	Legrafi Gábor

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat I. / Szakdolgozat II. (BSc) vagy Diplomamunka 1 / Diplomamunka 2 / Diplomamunka 3 / Diplomamunka 4³ tantárgy követelményét teljesítette, beszámolóra / védésre⁴ bocsátható.

Budapest, 2021. 12. 15.

Legrafi Gábor
intézményi konzulens

¹ Megfelelő aláhúzendő!

² Megfelelő aláhúzendő!

³ Megfelelő aláhúzendő!

⁴ Megfelelő aláhúzendő!

Tartalomjegyzék

1.	Bevezetés.....	8
2.	Jelenlegi piac elemzése	9
2.1.	SAP és SAP Business One.....	9
2.2.	Vector	11
2.3.	REVOLUTION – deep.erp	13
2.4.	Piac összegzése.....	14
3.	Fejlesztői dokumentáció.....	14
3.1.	A választott platform oka	14
3.2.	Specifikáció	16
3.3.	Use Case-ek meghatározása.....	17
3.4.	Szekvencia diagram	18
4.	Rendszerem adatbázisai	19
4.1.	A rendszer OLTP adatbázisa	19
4.2.	OLAP rendszerek felépítése.....	24
4.2.1.	A csillag séma	25
		25
4.2.2.	A galaxis séma	26
4.2.3.	A hópehely séma	26
		26
4.3.	A rendszerem adattárháza	27
5.	Felhasználói dokumentáció	30
6.	Szakdolgozat összegzése	38
7.	Summary.....	38
8.	Irodalomjegyzék	39
9.	Ábrajegyzék	40

1. Bevezetés

Eddigi munkáim során rengeteg alkalommal fordultak hozzám, hogy valamilyen kérdésre próbáljak meg választ adni. Legyen szó akár egy egyszerű profit kalkulációról, vagy éppen egy nagyon bonyolult és részletes KPI riportról, minden egyes alkalommal felmerült bennem a kérdés, hogy miért nem élnek több anyagi ráfordítással, vagy épp adnak nekünk elegendő időt arra, hogy egy olyan rendszert lefejleszthessünk, ami ezekre könnyedén választ adna, és rengeteg pénzt spórolna hosszútávon azáltal, hogy nem csupán az asszisztensek munkáját, hanem a menedzsment munkáját is jobbra tehetnénk ezekkel. Emiatt kezdett el foglalkoztatni, és sok esetben felajánlottam, hogy munkaidőmön belül megcsinálom, csak allokálják erre az időmet, de fontosabb volt, hogy inkább egyéb, kevésbé lényeges feladatok elkészüljenek, mintsem egy olyan rendszer, amiből profitálhatna is a cég a jövőben.

Ekkor kezdtem el azután kutatni, hogy mennyi olyasfajta rendszer lehet a piacon, ami hasonló dolgot nyújt, mint amit én kigondoltam. Sokat találtam kisebb-nagyobb cégek által, de mindegyikre igaz volt, hogy csak részben nyújtotta azt, amire akkor úgy igazán szüksége lett volna a cégnek. Ezen felül ezekért a rendszerekért, illetve szolgáltatásokért komoly összeget kértek a szolgáltatók, ami mellé sok esetben support díj is társult, úgyhogy a cégnek nagyon is a zsebébe kellett volna nyúlnia, ha szerette volna ezeket igénybe venni. Nem mellesleg, sokról köztudott volt, hogy hiába a support díj, nem ügyfél, hanem profitorientáltak.

Számomra fontosnak tartott dolgok között vannak a könnyen riportálható bevételi adatok, mint a nettó/bruttó költségek, a munkabér és egyéb kiadások, illetve nyilván a profit. Ezeket leszámítva a cégvezetés számára fontos, hogy legyenek terveik azokkal kapcsolatban, hogy miket-hogyan szeretnének elérni, vagyis úgynevezett „target” számai. Ezek a számok megmondják, hogy melyik vezetőnek milyen eredményeket kellene hoznia, illetve fontosak valamilyen tervezett összegek is munkák szempontjából, hogy a jövőbeli költségekkel és bevételekkel hónapokkal előre tervezni lehessen. Későbbiekben ezekből könnyen lehet terv/tény riportokat készíteni, melyeket, ha idővel sikerül jól meghatározni, egy cég vezetése nagyon könnyen tudna pénzügyileg előre tervezni olyan dolgokkal kapcsolatban, mint bevételek és kiadások.

Ezek a számok persze „önmagukban” mit sem érnek, hiszen egy cégvezetés nem ezeket fogja nézni, hanem az ezekből kialakított vizualizációs modelleket, amik látványos diagramokon megmutatják, hogy melyik üzletág miként teljesített, és miért. Ezek kulcsfontosságúak egy terv és riport készítő rendszer tökéletes működéséhez.

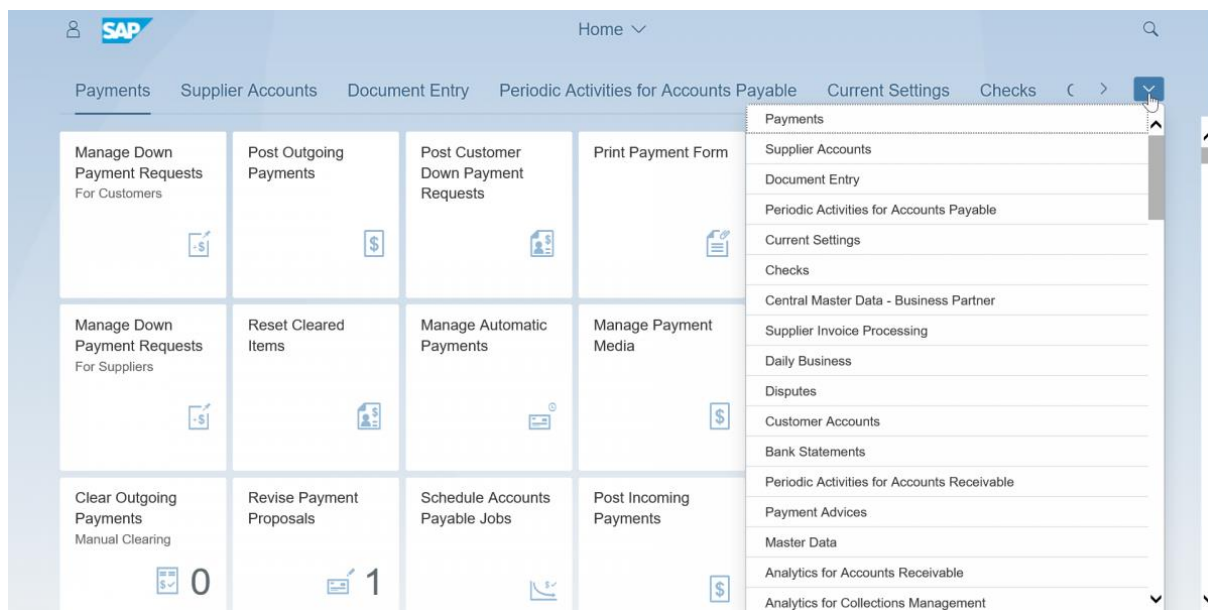
Mindezeket összegezve, céloom egy olyan tervező és riport készítő rendszer (egyfajta kontrolling rendszer), ami egy vezetés számára könnyű útmutatást adhat azzal kapcsolatban, hogy milyen számokat voltak képesek hozni profit terén, illetve előre tervezhetnek könnyedén, hogy egyes munkákra a jövőben mennyi időt kell pozíciónként allokálni. Ezáltal a menedzsment egy tisztább képet kaphatna arról, hogy milyen számokat várhatnak a jövőben. Amennyiben ez nem olyan számokat mutat, amit a vezetőség vár, könnyedén új terv készíthető, ezáltal könnyebb rálátást nyújt a hosszútávú költségek lefaragásában.

2. Jelenlegi piac elemzése

Rengeteg hasonló kontrolling/riportáló rendszer létezik a piacon, viszont szinte mindegyik azonos problémákkal küzd. Tapasztalatom alapján sok esetben egy „dobozos” terméket kínálnak, ami általánosságban jó lehet a cégeknek, de senkinek sem elégíti ki teljességgel az igényeit. Probléma velük, hogy sok esetben lassúak, nem optimalizáltak, és néha még elavultak is, mindezekért pedig komoly összegeket elkérnek a piacon a szolgáltatók (érthető, hisz ettől függetlenül egy nagyobb, komolyabb cég számára egy ilyen nélkülözhetetlen). A szoftver alap költségén felül még a support díjakkal is számolni kell, hisz a felmerülő problémákat mihamarabb meg szeretnénk oldani, viszont mindazok ellenére, hogy a cég ezért a zsebébe nyúl, gyakran nem azt kapja a szervezet, amire teljes mértékig szüksége volna. Nézzünk néhány példát a piacon található szoftverekre.

2.1. SAP és SAP Business One

Kezdjük is talán az egyik legnagyobb vállalatirányítási rendszerrel, mégpedig az SAP-val. Az SAP rengeteg ágazatban megtalálható, ilyen például a pénzügyi szektor, vagyis a banki szféra, illetve biztosítások, energia és természeti erőforrások, valamint a fogyasztói iparág. Mindegyik téren egyedi intelligens technológiák által vezérelt szolgáltatásokat kínál. A cél, hogy a vállalatok hatékony működését lehetővé tegyék. A cég több, mint 120 országban jelen van, és megközelítőleg 32 ezer vállalat használja aktívan, a kis és középvállalkozásokon át egészen a nagyvállalatokig. Többféle terméket kínálnak, például adnak dobozos termékeket is, mely az SAP Business One vagy az SAP R/3, illetve legújabb termékük az SAP S/4 HANA, ami már a cég saját in-memory adatbáziskezelőt használja.



2.1. ábra az S/4 HANA kezelőfelete

Az SAP Business One-t 2002-ben indította útjának az SAP. Legfőképp kisvállalatoknak ajánlja a cég. Ez a rendszer lehetőséget nyújt a cég ellenőrzésére, így az üzleti folyamatok könnyen összeköthetőek lesznek, ezáltal hatékonyabbá válik ezen vállalatok ellenőrzése. Alap célja, hogy kulcsfontosságú folyamatokat egyszerűsítsen, ezáltal könnyebb bepillantást enged a cég mindennapjaiba, ez pedig a vállalat vezetőségének számára egyszerűbb döntéshozatalt eredményez. A rendszer moduláris felépítéséből kifolyóan képes a változó igények kielégítésére könnyűszerrel, ugyanis bármikor bővíthetőek a licenszelt modulok. A cég sokféle ágazatban próbál a vállalatoknak informatikai szolgáltatást nyújtani. A továbbiakban elemezni fogom ezen szolgáltatások közül a legfontosabbakat.

Kezdjük a pénzügyi menedzsment fejlesztésével. Célja ennek, hogy a cégek a pénzügyi helyzetüket könnyedén átláthassák, ezáltal javíthassák úgy, hogy automatizálják a könyvelési folyamatokat, illetve egyéb pénzügyi műveleteiket egységesítik. A szoftver a gondoskodik a teljes könyvelési feladatok és folyamatok lefedéséről, kezeli a pénzáramlást, a tőkét, és monitorozza a felhasználó vállalkozásunk számára, hogy milyen költségekkel kell még számolnunk, mindezt nagy pontossággal teszi. Mindezekon felül képes leegyszerűsíteni az adatbetöltő folyamatokat, amik a folytonos, manuális betöltések már nem lesznek szükségesek. Ezen szoftver segítségével a pénzügyi feladatokat ellátó munkatársak alacsonyabb hibaszázalékkal dolgozhatnak.

Az SAP Business One másik jelentős részlege az üzleti intelligencia. Ennek célja az adatbetöltés, a rendszeres riportálás, melyek könnyen konfigurálhatóak a cég igényeitől függően. Ezen kívül időszakos jelentéseket is készít a szoftver a cég számára a meglévő

nagymennyiségű adatokból, különféle analitikai és riportáló eszközök felhasználásával. Ezek a jelentések nem feltétlen egy helyről kell származzanak, több forrásból is jöhetnek, illetve meglévő jelentések is bármikor módosíthatóak könnyedén. Az excel integritásnak köszönhetően könnyedén lehet interaktív analíziseket folytatni.

Az utolsó, általam fontosnak vélt rétegük az a termékeket és ügyfeleket kezelő rendszerük. Ez lényegében azért felelős, hogy az összes vásárlási folyamatot és vásárlást felügyelje, ezen kívül az ügyfelek lehetséges jövőbeli vásárlásait kalkulálja. Ezzel a cégek könnyedén képesek követni a teljes életciklusát egy terméknek, és könnyen analizálhatják, hogy milyen változások várhatóak a piacon, mindezt a profitmaximálás érdekében.

Látható, hogy ezen szolgáltatások miatt, és nagy rugalmasságuknak köszönhetően, ami a dobozos szoftver előnyeiből származik, sok vállalkozásnak segíthet a mindennapi menedzsmenti feladatok eldöntésében az SAP. [1]

2.2.Vector

Egy teljesen magyar tulajdonú cég, ami 1989-ben alakult, vagyis már 30 éve vállalatirányítási rendszer fejlesztésével foglalkozik. A cég folyamatosan fejleszti, s naprakészen tartja szoftverét, ezáltal pedig ma már több, mint 500 cég használja, ezzel pedig 14 ezer fős személy dolgozik az általuk nyújtott rendszerekkel. Első körben a hazai vállalatoknak próbál kedvezni. Rendszerük moduláris, így teljes egészében az ügyfél igényeire szabható, és igyekeznek az ügyfélközpontú arculatot kialakítani, így bevezetéstől kezdve már a partnerek mellett vannak. A rendszer hatalmas modulválasztékkal rendelkezik, így bármely cég képes megtalálni a számára szükséges szolgáltatást. Főbb moduljaik közé tartozik a webáruházak, beszerzések és értékesítések, pénzügy, kontrolling, valamint a projektmenedzsment.

Kontrolling moduljuk segítséget nyújthat üzleti célú döntések meghozatalában, hiszen a menedzsment pontosan látni fogja, hogy melyik az a kiugró érték, ami nem várt eredményt hozott. A modul részét képezi a lekérdezések automatizálása, így a vezetőségnek megadott időpontokban a rendszer kiküldi a kért jelentést olyan formában, ahogy az épp szükséges. Ezen kívül részletes kimutatást nyújt a számok/értékek alakulásáról, a tervezőmodullal kiegészítve pedig egy kisebb tervezőrendszerre alakul, amivel különféle terveket készíthetnek az illetékesek, melyeket később összevethetnek a tényadatokkal. A terv adatok azért is számítanak fontosnak, mert jó tervezés esetén a cég felkészülhet egy-egy komolyabb kiadásra, illetve könnyedén módosíthat a tervén, így képes lesz költséghatékonyabban elvégezni azt az adott feladatot.

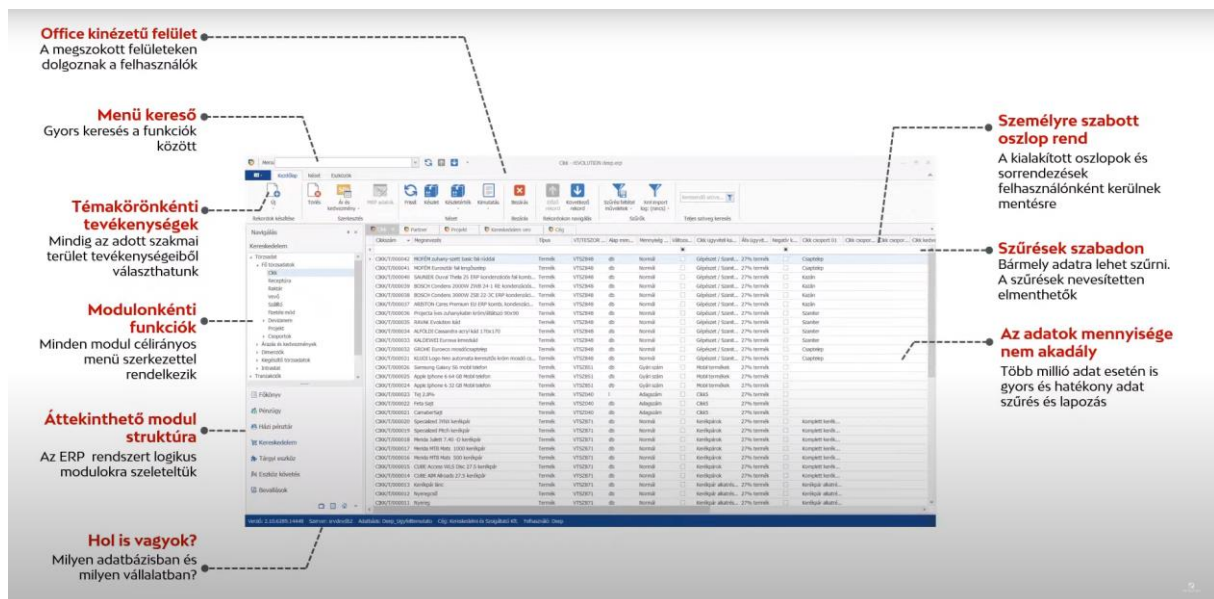
Másik nagyobb moduljuk a pénzügyi és számviteli modul. A rendszer pontos képet ad arról, hogy milyen pénzügyi változások történtek, ezen kívül naprakész cash-flow és számla információkat jelenít meg. Természetesen ezen kívül rengeteg banki interfész részét képezi, így könnyen tud küldeni és fogadni utalásokkal kapcsolatos adatokat. A könyvelés egy nagy része ennek a rendszernek, mely vevő és szállítói számlákat, árfolyam nyereséget/veszteséget, eszközmozgásokat, készletet vagy épp amortizációt rögzít és kezel. Ezen kívül könnyedén készít kimutatásokat és jelentéseket könyvelési tételekről. Képes kezelni elektronikus számlákat, így a vevő elektronikus úton is könnyedén hozzájuthat számlájához. Ezek mind a Nemzeti Adó és Vámhivatal által előírt formátumot követik.

A projektmenedzsment egy kisebb, ám lényeges része a nagy egésznek. Ezzel a cégek könnyedén figyelemmel kísérhetik projektjeiket, azok feladatait és tervezhetnek, kalkulálhatnak velük. Minden adott ahhoz, hogy ezeket könnyedén megtehessék, ugyanis ez a modul lehetőséget biztosít a projektek dokumentálására, a feladatok könnyűszerű megtervezésére, majd allokálására, az eddigi eredmények megtekintésére, ezekből pedig konzekvenciák levonására. Végezetül, projektek lezárására és ezek kiértékelésére is van lehetőség. [2]

2.3. REVOLUTION – deep.erp

A REVOLUTION Software Kft. vállalatnagyságtól függetlenül ajánlja rendszerét, ha az adott cégnek vállalatirányítási rendszerre van szüksége. 30 éves tapasztalat és 400 projekt jellemzi a céget. A szoftver kínálatuk széleskörű, így sok cégnek tudnak megfelelni.

A deep.erp-t vállalatok folyamatainak gyorsítására fejlesztette a cég, ezeket pedig különféle automatizációk által kívánja elérni. Próbálnak a paramétereizhetőségre törekedni, így csökkenthetőek a felállítás és a megvalósítás költségei. A rendszeren belül könnyedén belső kommunikáció folytatható az üzenetküldő rendszerüknek köszönhetően és feladat kiosztó moduljuk révén egyszerűen kioszthatóak a munkahelyi feladatok.



2.2. ábra deep.erp kezelőfelülete

ERP rendszer révén könnyedén testreszabható, egyedi fejlesztést pedig nem feltétlen igényel paramétereizhetősége miatt. Elmondásuk szerint a rendszer automatizációi alkalmazkodnak a cég igényeihez, ezáltal csökkenthetőek vagy akár meg is szüntethetőek az ezekkel járó repetitív, hosszú tevékenységek, ezzel pedig erőforrást szabadítunk fel. Moduljaik részét képezik a pénzügyi, kontrolling, illetve készletkövető moduljaik és még sok más. Pénzügyi moduljuk segíthet a cég pénzügyi helyzetéről friss információkat nyújtani, legyen szó bármely pénzügyi ágról a vállalatban belül. A cég banki kivonatairól, bizonylatairól és egyéb könyveléssel kapcsolatos folyamatairól információt nyújthat ez a modul. Ez kezeli a főkönyvi tételeket is, mely magába foglalja az ÁFA analitikát is.

A vezetői információs rendszer modulja jelentéseket képes megjeleníteni, melyek bármikor exportálhatóak Excelbe. A modul nem követeli meg, hogy bonyolult, költséges kimutatásokat készítsenek előtte, ezeket könnyedén megtehetik a rendszert használók egy egyszerű adattáblában. [3]

2.4. Piac összegzése

Látható, hogy mennyire sokoldalú a piac. Sok-sok cég kínál rengetegféle megoldásra problémát, egymás ilyen-olyan módon átfedve, de kicsit egyedi módon. Valaki felhővel, valaki az ügyfélközpontúsággal akar minél több bevételre szert tenni. Mindegyik megoldás jó a maga módján, viszont az én rendszeremnek két kiemelkedő része lesz, ez a kettő pedig a tervezés (mind bevétel, mind kiadás terén) a másik pedig a terv és tény adatokból alkotott különféle jelentések készítése. Egyik rendszer sem képes ilyen mélységig mindkettőre, általában inkább csak az adatvizualizációra fókuszálnak, azonban én ennél többre töreksem. Egyik rendszer sem fókuszált teljes egészében erre a két feladatra, így ezek nem volnának számomra megfelelőek.

3. Fejlesztői dokumentáció

3.1. A választott platform oka

Az általam fejlesztett rendszer az alábbi négy fő részből fog állni, ami nem más, mint egy tranzakció alapú adatbázis, egy adattárház a jelentések készítéséhez, a backend, ami a frontend és az adatbázis-szerverek közötti kommunikációt fogja megvalósítani, illetve maga a frontend, ami a vizualizációért és a tervezésért lesz felelős. A rendszer egy webes szoftver lesz, aminek okát lejjebb a frontend szakaszban részletesen ki is fejtem.

Az adatbázist egyértelműen relációs alapokon kell véghez vinnem, elvégre szeretnék ebből jelentéseket készíteni egy adattárház felhasználva. Természetesen NoSQL is használható lenne az OLTP rendszer esetében, például egy mongoDB a dokumentum alapú felépítésével, mely a nyers adatokat tartalmazhatná a terv és tényadatokról, de azokat is jobbnak látom relációs alapokon megcsinálni, ugyanis egyszerűbb lesz létrehozni az OLAP rendszerhez szükséges adattárház struktúráját. Az adatbázisom Microsoft SQL Server lesz. Lehetne PostgreSQL is akár, de a rendszerem nem lenne képes kihasználni azokat az előnyöket, amiket ez az adatbázis-szerver nyújtani tudna, vagy akár Oracle, illetve MariaDB, de egyik adatbázist sem ismerem olyan szinten, mint az SQL Servert. Szerencsére, ha a jövőben változtatnom kéne, a backendem úgy lesz kialakítva, hogy könnyedén tudok szerverek között ugrálni az Entity Framework Core kialakítása végett.

Az adattárház alapjait szintén az SQL Server fogja nyújtani. Nem az összes táblám lesz felhasználva, ezeket optimalizálom az adattárházam struktúrájára, mivel így gyorsabban fogja tudni a rendszer a kért jelentéseket elkészíteni. Egy csillagséma lesz az alapja, aminek a tény táblája a munkatábla lesz. Az adatok betöltéséért néhány tárolt eljárás fog felelni, amik beállíthatóak lesznek később, hogy milyen rendszeresen legyenek futtatva. Az adattárházam tartalmazni fog SCD Type 1 és Type 2-es táblákat is.

Backend esetén mindenképp a Microsoft által létrehozott C# .NET keretrendszert fogom használni egy rest API-t létrehozásához, ami az adatbázissal való kommunikációért fog felelni. Verziót tekintve .NET-en belül is a jelenlegi legújabb LTS verziót, mégpedig a .NET Core 5.0-es verziót fogom választani. Szakdolgozatom írásakor már az .NET Core 6.0-s verziónak az rc2-es verziója érhető még csak el az interneten, de véleményem szerint az 6.1-ig nem érdemes foglalkozni vele, ugyanis hiába tartalmaz rengeteg újítást, egy új verzió mindig tartalmaz új, nem várt hibákat is, amiket feltehetően 6.1-es verzióban már javítani fognak. Ezen felül a backendem szerves részét fogja képezni az Entity Framework Core ORM rendszer, amivel a rendszer CRUD műveleteit könnyedén implementálhatom.

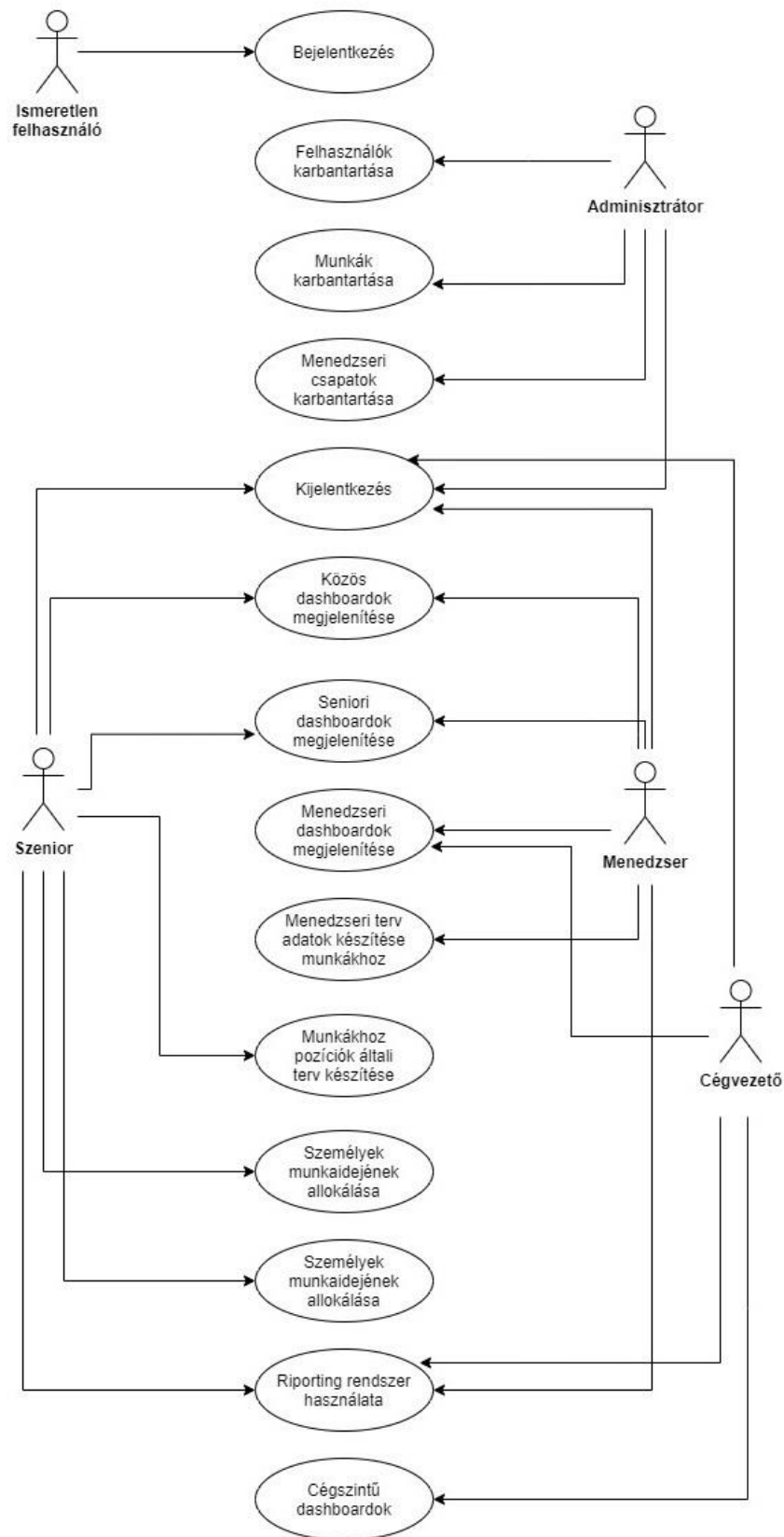
Frontend terén sokáig egy asztali és mobil projektben gondolkoztam, viszont végül elvetettem, ugyanis úgy vélem, ilyen rendszerre nem szükséges egy mobil alkalmazás, míg egy asztali alkalmazás lehet egyes cégektől túl nagy elvárás volna, hiszen nem szabad elfelejteni, hogy az ügyfelet nem lehet bekorlátozni azzal, hogy megszabjuk, miféle szoftvereket tegyen fel a rendszerre csak azért, hogy az általunk létrehozott rendszer futtatható legyen. Ilyesfajta lehetőségeket a cross-platform tud nyújtani és egy webes alkalmazás, hiszen ennek semmi másra, csak hálózatra és egy böngészőre van szüksége. Emiatt a frontendemet a Google által fejlesztett Angularral 12-vel fogom megvalósítani, ami az egyik jelenlegi nagy Javascript Framework manapság a piacon. Ráadásul könnyen lehet optimalizálni a nézeteket mobilra egy webes alkalmazásnál, így ha szükség volna rá, a jövőben könnyedén mobilra optimalizálható lenne. Használhattam volna ASP.NET MVC-t is akár, vagy PHP-t a frontend elkészítésére, de ahhoz ugyanúgy szükség lett volna Javascriptre a megfelelő és gördülékeny felhasználói élmény miatt, ezáltal egyszerűbbnek tartottam, hogyha ezt egyetlen rendszerben megoldom. Vállalati környezetben SPA-k (single-page application) amúgy is elterjedtek egyszerűségük és gyorsan fejleszthetőségük miatt.

3.2. Specifikáció

A rendszernek az alábbi követelményeket teljesítenie, míg funkciók terén az összeset tartalmaznia kell.

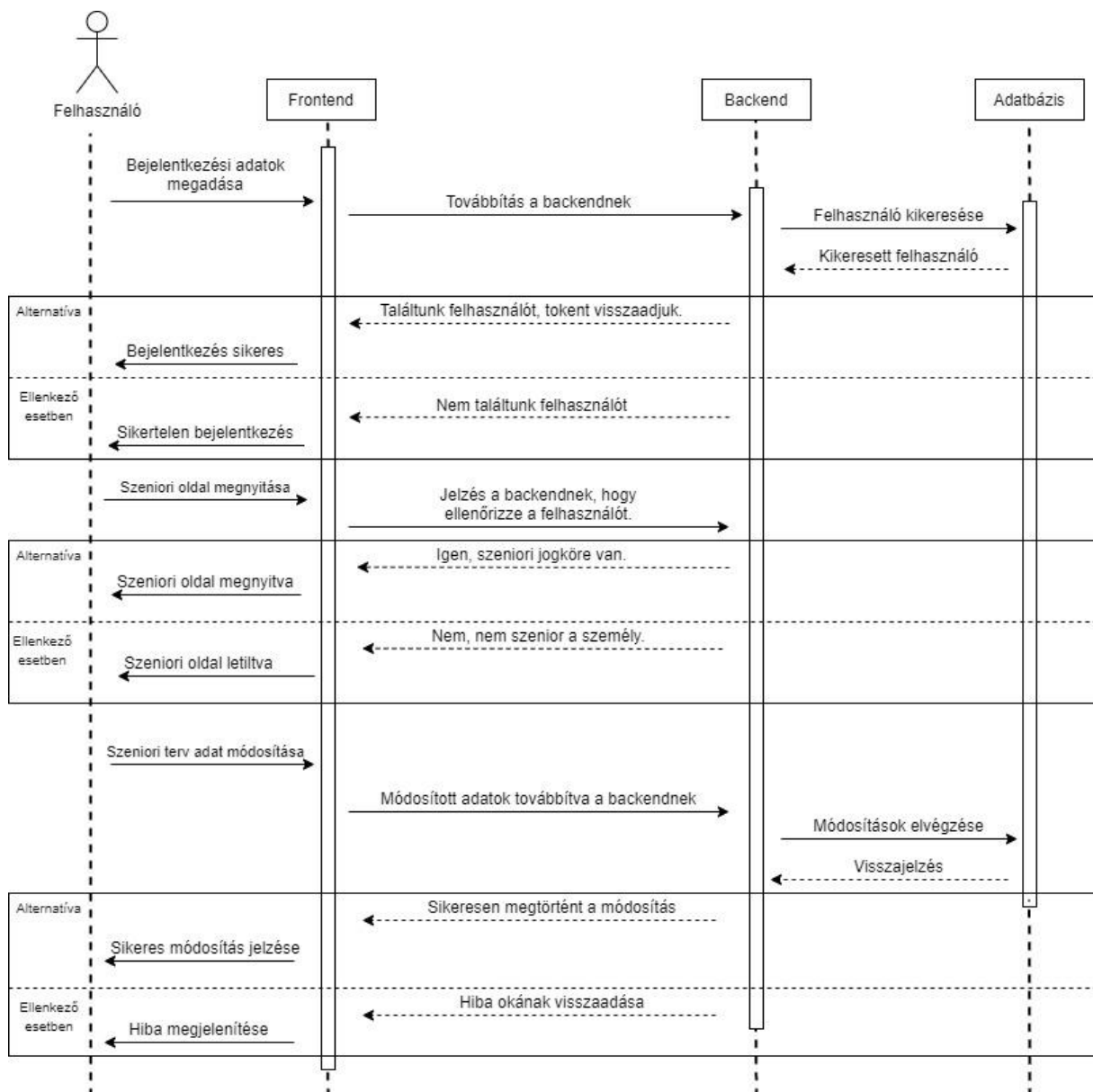
- Új felhasználók bármikor hozzáadhatóak a rendszerhez, vagy épp deaktiválhatóak. Ehhez egy külön adminisztrációs oldal lesz elérhető, amit csak egyes személyek érhetnek el a megfelelő jogosultságokkal.
- Szükséges egy jogosultságrendszer, ami alapján dől el, ki-milyen dashboardokat lát, illetve milyen oldalakra léphet be. A jelentések is le lesznek szűrve, lesz olyan, amit minden szenior/menedzser láthat, de lesznek jogosultságspecifikus jelentések is.
- Egy szenior előre megtervezheti véleménye szerint egyes munkákon, egyes pozíciók mennyi időt fognak eltölteni heti szinten.
- A rendszer egyedi naptárrendszert kell használjon, így ugyanis könnyen eldönthető, hogy egy hét melyik hónaphoz fog tartozni, ha esetleg 2 hónapba is belelóg.
- Minden adatot heti szinten kezelünk (terv és tény adatokat egyaránt) a könnyebb tervezhetőség és átláthatóság miatt.
- Munkákra lehet tervezni, hogy mely pozíció mennyi időt tölthet vele hetente átlagosan.
- Egy munkát a szenior a csapatában képes konkrét személy(ek)re allokálni (ki-mennyit-min fog dolgozni egy adott héten).
- Egy asszisztens/senior több menedzser alatt is dolgozhat. Ezeket meg kell tudni határozni, hogy melyik személy melyik menedzser alatt van, és hány órában.
- A rendszerben a könnyebb átláthatóság érdekében sok szűrési feltétel lesz elérhető, így munkákra, személyekre, menedzserekre/szeniorokra és természetesen időszakra is lehet majd szűrni.
- A rendszer a terv adatokat egy OLTP rendszerben fogja tárolni.
- Szükség lesz riporting rendszerre, ahonnan az összes jelentése elérhető lesz. Ezeknek mind szűrhetőnek kell lennie, ráadásul mindenki csak a saját adatait láthatja ezekben a jelentésekben.
- A riporting rendszer alapját egy OLAP rendszer fogja nyújtani, ami az OLTP alapjára lesz kiépítve. Ennek oka a gyorsabb riportálás lesz.
- Ehhez a riporting rendszerhez szükséges lesz egy adattárház a nagymennyiségű adat miatt.
- Minden dashboardon található táblázat letölthető kell legyen excelként vagy .csv-ként.
- A rendszernek teljes egészében weben kell futnia.

3.3. Use Case-ek meghatározása



3.1. ábra A rendszerem használati eset diagramja

3.4. Szekvencia diagram



3.2. ábra A szeniori szerepkör szekvencia diagramja

A fenti diagramon a szeniori szerepkör szekvencia diagramja látható. A weboldal használatához kötelező belépnie a felhasználónak, így ez természetesen azon oldalakra is igaz, amelyeket szeniori jogkörre lettek korlátozva. Ez alapján először is a felhasználónak meg kell adnia a belépési adatait a frontenden, ami utána továbbítva lesz a backend felé. Ezen adatokat felhasználva a backend az adatbázisból megpróbálja kikeresni a felhasználót. Ha találunk felhasználót, akkor bejelentkeztetjük, ellenkező esetben sikertelen bejelentkezés történt, és ezt jelezzük a felhasználó felé.

Ezek után bármely oldalt megnyitjuk, ami szeniori jogkörrel látogat, ellenőrizzük, hogy az adott felhasználó valóban rendelkezik azon jogosultságokkal, hogy láthassa az adott oldalt. Ha nem, visszaadjuk, hogy az oldal letiltva, nem látható a felhasználó számára, ellenkező esetben pedig megnyitjuk a kívánt oldalt.

Adatmódosítás esetén a frontenden elvégezzük a kívánt módosításokat (hozzáadás/módosítás/törlés), majd ezen adatokat továbbítjuk a backend szervernek. Ez elvégzi a szükséges módosításokat, majd az adatbázis szervernek továbbítja, hogy ezen módosításokat végezd el. Arról kap a backend egy visszajelzést, ami alapján vissza tudja adni a felhasználónak, hogy a módosítás, amit elvégzett, megtörtént-e avagy sem. Ha megtörtént, jelezzük a sikeres módosítást, ellenkező esetben jelezzük, hogy valami hiba történt.

Ez a logika az összes jogosultság esetén így fog működni, leszámítva a cégvezetőnél, ugyanis ennek a szerepkörnek csak olvasási jogköre van, semmilyen adatot nem módosít a rendszeren belül.

4. Rendszerem adatbázisai

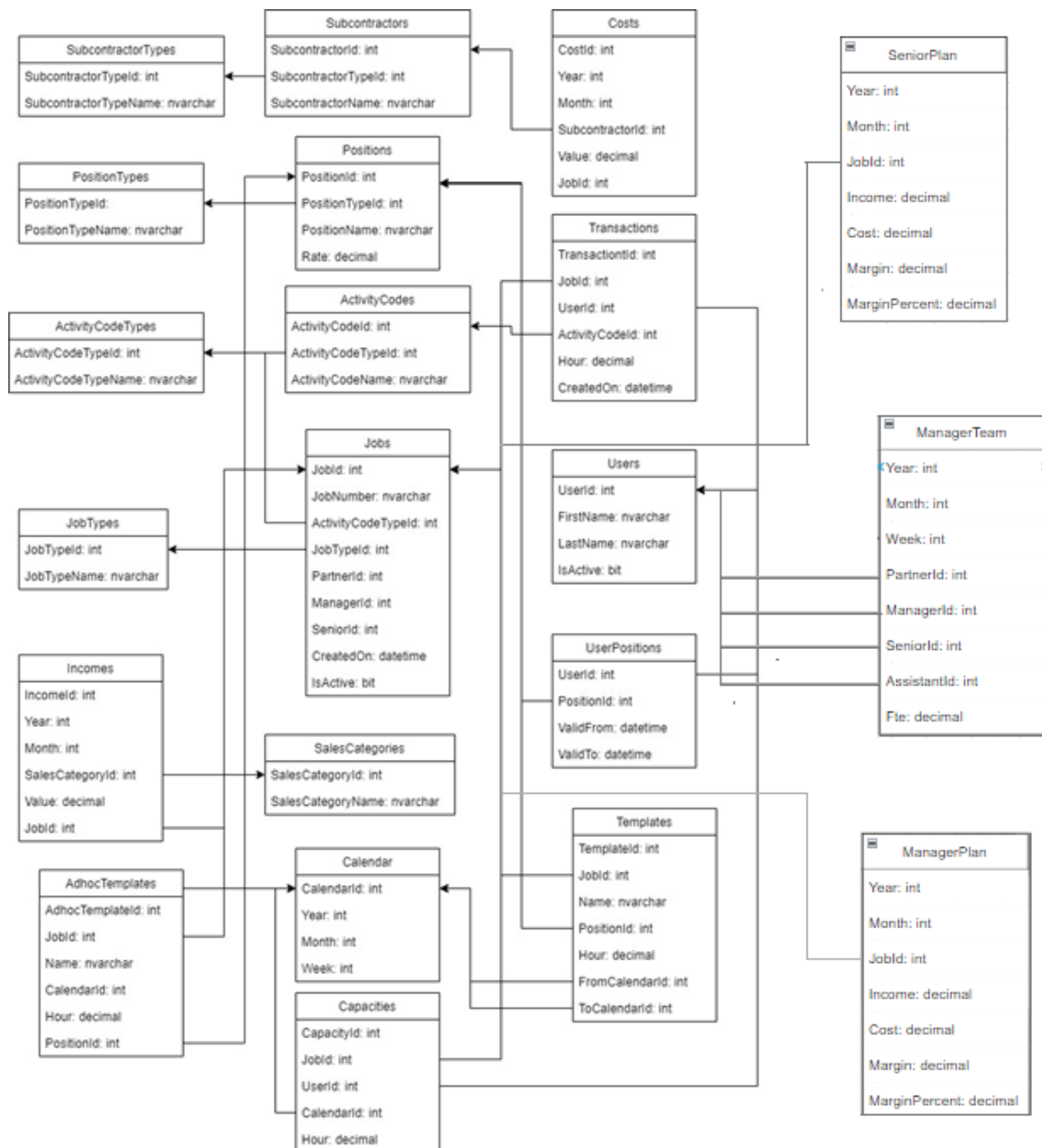
4.1. A rendszer OLTP adatbázisa

Miután eldöntöttem, hogy a rendszer alapját relációs adatmodellben képzeltem el, végig kellett gondolnom, hogy miként is alakítom ezt ki. Mivel a rendszerem több táblát fog tartalmazni, ráadásul sokba írni is lehet, így természetes, hogy az adatbázis alapját egy OLTP rendszer fogja adni. Egy OLTP (Online Tranzakciófeldolgozás) rendszer nem más, mint egy tranzakciós adatfeldolgozó rendszer. Arra jó, hogy üzleti szervezet napi szintű változásait könnyűszerrel nyomon kövesse. Egy ilyen rendszer megkövetel néhány dolgot annak érdekében, hogy biztosítani tudja, hogy minden tranzakció megfelelő módon futott le a rendszerben.

Az egyik az, hogy minden tranzakciónak atominak kell lennie, vagyis minden egyes tranzakcióra igaznak kell lennie, hogy vagy teljesen sikeres volt a tranzakció, vagy teljesen sikertelen, nincs köztes állapot. Ha egy tranzakció a művelet felénél hibára fut, s nem hajtható teljesen végre, az adatbázis-szervernek vissza kell állnia arra az állapotra, ami még a tranzakció indítása előtt volt, így biztosítható, hogy semmilyen adatvesztés sem történt a rendszerben. Ezek az úgynevezett „rollback”-ek az adatbázis által automatikusan megtörténnek az adatbázisban, de egy adatbázis-fejlesztő is bármikor használhatja ezeket a munkája során. Az adatbázisok különféle zárolási stratégiákat használnak arra, hogy biztosítani tudják, hogy minden egyes tranzakció, amit a vállalat összes dolgozója elindított, képes legyen gördülékenyen és hibamentesen, összeakadás nélkül lefutni.

Egy ilyen rendszer hátránya viszont, hogy nagyobb mennyiségű adat esetén a lekérdezések lassúak lehetnek, főleg bonyolultabb lekérdezések esetén, hiszen nem erre lett optimalizálva, hanem adatok felvitelére, módosítására, illetve törlésére. A rendszernek viszont szüksége lesz bonyolultabb jelentésekre, amik ennyi adatnál problémát okozhatnának egy ilyen adatbázisnak, így egy adattárház is része lesz a rendszernek. Az adattárházról részletesebben írok a következő menüpontban, most viszont nézzük meg, hogyan is néz ki az OLTP adatbázisom struktúrája.

[4]



4.1. ábra A rendszerem OLTP adatbázisának szerkezete

A következő főbb táblákat használja az adatbázisom:

Jobs (tartalmazza a munkákat):

- JobId: tábla azonosítója
- JobNumber: egy szöveges egyedi kód
- ActivityCodeTypeId: a feladattípus tábla egyik azonosítója
- JobTypeId: a munka típusa tábla egyik azonosítója
- PartnerId: a munkát irányító cégvezető idegen kulcsa
- ManagerId: a munkát irányító menedzser idegen kulcsa
- SeniorId: a munkát irányító szenior idegen kulcsa
- CreatedOn: a rekord létrehozásának dátuma
- IsActive: Aktív-e a rekord

Transactions (tartalmazza a dolgozók által regisztrált munkaórákat):

- TransactionId: tábla azonosító
- JobId: a munkák tábla idegen kulcsa
- UserId: a felhasználók tábla idegen kulcsa
- ActivityCodeId: a feladatok tábla idegen kulcsa
- Hour: a munkán töltött órák száma aznap
- CreatedOn: A munkán töltöt idő pontos napja

Incomes (tartalmazza a céges bevételek hónapra összegzett értékét):

- IncomeId: tábla azonosító
- Year: melyik évre vonatkozik
- Month: melyik hónapra vonatkozik
- SalesCategoryId: a bevételi típusok tábla idegen kulcsa
- Value: a bevétel összege
- JobId: a munkák tábla idegen kulcsa

Costs (tartalmazza a céges kiadásainak hónapra összegzett értékét):

- CostId: tábla azonosító
- Year: melyik évre vonatkozik
- Month: melyik hónapra vonatkozik
- SubcontractorId: a bevételi típusok tábla idegen kulcsa
- Value: a bevétel összege
- JobId: a munkák tábla idegen kulcsa

ManagerTeam (tartalmazza a rendszerben használt csapatokat heti bontásban):

- Year: melyik évre vonatkozik
- Month: melyik hónapra vonatkozik
- Week: melyik hét vonatkozik
- PartnerId: a partner azonosítója
- ManagerId: a manager azonosítója
- SeniorId: a senior azonosítója
- AssistantId: az asszisztens azonosítója

Users (tartalmazza a rendszer felhasználóit):

- UserId: tábla azonosító
- FirstName: a felhasználó keresztnéve
- LastName: a felhasználó vezetéknéve
- IsActive: a felhasználó még a cégnél dolgozik-e

Templates (tartalmazza a pozíciókra készíthető tervadatokat):

- TemplateId: tábla azonosító
- JobId: a munkák tábla idegen kulcsa
- Name: a terv adat elnevezése
- PositionId: melyik pozíciót érint ez a terv
- Hour: a tervezett óra hetenként
- FromCalendarId: a terv kezdetének hete
- ToCalendarId: a terv utolsó hete

AdhocTemplates (tartalmazza a pozíciókra készíthető adhoc tervadatokat):

- TemplateId: tábla azonosító
- JobId: a munkák tábla idegen kulcsa
- Name: a terv adat elnevezése
- PositionId: melyik pozíciót érint ez a terv
- Hour: a tervezett óra hetenként
- CalendarId: a terv konkrét hete

Capacities (tartalmazza a dolgozókra heti szinten kiszabott munkaórákat):

- CapacityId: tábla azonosító
- JobId: a munkák tábla idegen kulcsa
- UserId: a felhasználók tábla idegen kulcsa
- CalendarId: a kiszabott munka hetének dátuma

Calendar (tartalmazza az egyedi naptárrendszert, amit a rendszer használ):

- CalendarId: tábla azonosító
- Year: melyik évre vonatkozik
- Month: melyik hónapra vonatkozik
- Week: melyik hétre vonatkozik

UserPositions (tartalmazza a felhasználók pozícióit historikusan tárolva):

- UserId: a felhasználók tábla egyedi kulcsa
- PositionId: a pozíciók tábla egyedi kulcsa
- ValidFrom: mettől érvényes a pozíció
- ValidTo: meddig érvényes a pozíció

Positions (tartalmazza a lehetséges pozíciókat és azok tervnél használt órabérét):

- PositionId: a tábla egyedi azonosítója
- PositionTypeId: a pozíció típusok egyik azonosítója
- PositionName: a pozíció neve
- Rate: a terv adatoknál nézett pozíció órabére

ManagerPlan (tartalmazza a managerek havi tervezett adatait):

- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Income: a tervezett bevétel
- Cost: a tervezett költség
- Margin: a tervezett profit
- MarginPercent: a tervezett profit százalék

SeniorPlan (tartalmazza a managerek havi tervezett adatait):

- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Income: a tervezett bevétel
- Cost: a tervezett költség
- Margin: a tervezett profit
- MarginPercent: a tervezett profit százalék

4.2. OLAP rendszerek felépítése

Az előző szakaszban kifejtettem részletesen, hogy miért van szükségem egy OLTP adatbázisra, most pedig részletezném, hogy miért van egy OLAP (Online analitikus feldolgozás) rendszerre is szükségem. Adattárházakra azért lehet szüksége egyes vállalatoknak, ugyanis a folyton csak növekvő, s növekvő adathalmazból egyre nehezebb lesz rövid időn belül bonyolult, összetett analitikai jelentéseket készíteni. A rendszert folyamatosan terhelik a beérkező tranzakciók, viszont egy ilyen jelentés elég komoly erőforrásigénnyel járhat, ami pedig lelassíthatja az egyes tranzakciók mentését, egészen addig, amíg a jelentés el nem készült. Erre fejlesztették ki az adattárházakat. Ezekbe az adatbázisokba áttölthetünk nagy mennyiségű adatokat ETL folyamatokon keresztül, felépítjük a megfelelő indexeket, kialakítunk egy megfelelő sémát, és onnantól kezdve az adatok analizálásához azt az adatbázist fogjuk használni. Nem szükséges csak a saját rendszerünk adatait felhasználni, könnyedén integrálhatunk több forrásból is.

Az adattárházak egy másik előnye, hogy általában történik egy új adatmodell kialakítás, ami esetén az adattáblákat úgy alakítjuk, hogy az könnyebben értelmezhető legyen a felhasználók számára. Példa erre, hogy összeépítünk 3-4 adattáblát, így egyszerűsítve a sémánkat. Ez azért segíthet, mert redukáljuk a kapcsolatok (joinok) számát az adattárházban, ezen felül nem kell a felhasználóknak a forrás adatbázist ismerniük, hisz ezeket van, hogy üzleti analizálók használják, ők pedig nem adatbázisfejlesztők.

Ezeket az új adatmodelleket kétféle szemantikai modellekbe sorolhatjuk: van a táblázatos és a többdimenziós. A táblázatos az a megszokott szerkezet, vagyis mint egy sima relációs adatbázis, táblákából és oszlopokból áll. A többdimenziós modell pedig a hagyományos OLAP-modellezési szerkezet. Ebbe tartoznak az adatkockák például.

Hátrányai közé sorolható, hogy az adattárházban lévő adatok nem frissülnek olyan gyakran, mint egy OLTP rendszerben. Ennek oka ugyebár, hogy általában nagyobb mennyiségű adathalmaz kerül be az adatbázisba, általában megadott időközönként. Ebből kifolyólag inkább stratégiai üzleti kérdésekre tud választ adni, nem képes a lehető legfrissebb adatokat felhasználva jelentéseket készíteni.

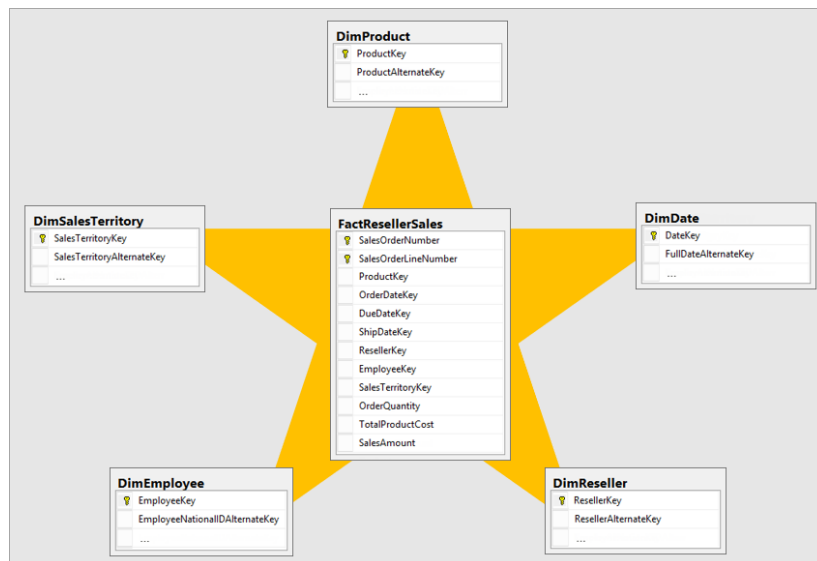
Ezen felül a többdimenziós működésükből adódóan nem könnyű, néha szinte lehetetlen az entitások közötti kapcsolat felállítása, hiszen ez a felépítés nem ad egy teljesen normalizált rendszert. Hogy ne legyen szükséges a normalizálás, különféle adatbázis sémák lettek kialakítva, ezzel segítve az adattárház struktúrák kialakítását. Ilyenek például a csillag, galaxis avagy a hópehely séma. Nézzük meg ezeket a sémákat, hogyan is épülnek fel. [5]

4.2.1. A csillag séma

A csillagséma nevezhető a legegyszerűbb data mart sémának. A séma alapját egy tény tábla adja, illetve több dimenziótábla, amik mind a tény táblához kapcsolódnak. A nevét a kialakítása adta, ugyanis a modell alakja egy csillagra emlékeztet. Ha a csillag sémánk tartalmaz egy táblát, ami az időt jelöli (mondjuk egy egyedi naptár tábla), akkor abban az esetben ezt a lehető legkisebb időegységet kell használnunk a teljes sémán belül, amit ez a tábla használ. Így garantálható, hogy az összes adat megfelelően lesz csoportosítva idő szempontjából. A következő részben elmagyarázom, mi egy tény és dimenziótábla között a különbség, és mik a különbségeik.

A tény táblák valamilyen eseménnyel kapcsolatos eseteket tárolják magukban. Sok esetben rengeteg dimenzióhoz kapcsolható idegen kulcsot tárol, hiszen ugye ezekben a dimenziótáblákban vannak az információk eltárolva. Ezek a táblák elég nagyok, hisz mindenféle lehetőséget magukban kell tárolniuk. Általában surrogate kulcsot tartalmaz, mint elsődleges azonosító, ugyanis mindegyik sornak egyedinek kell lennie még egy historikus adat esetében is, így ha a business kulcs ugyanaz is több rekord esetében, nem lesz problémánk egy konkrét rekord azonosítása.

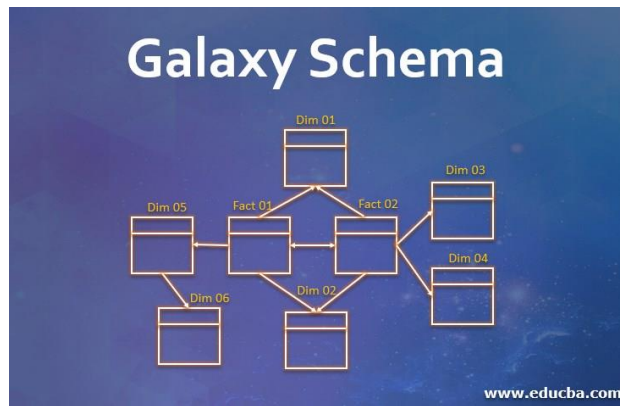
A dimenziótáblák kis rekordszámú táblák a tény táblához hasonlítva, de elég sok tulajdonságot magában foglal ahhoz, hogy a tény táblát teljességgel körül tudja írni. A dimenziótáblák lényegében az üzleti entitások, amik lehetnek személyek, eladások, vagy éppen a fent említett idő. [6] [7]



4.2. ábra Egy csillag séma ábrája

4.2.2. A galaxis séma

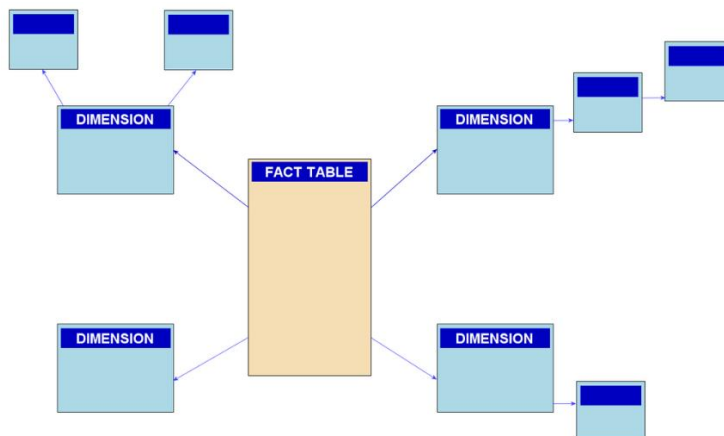
Következő sémánk a galaxis séma. A nevét a „többcsillagos” felépítéséből kapta, meg a kialakításától. A felépítés alapját több tény tábla nyújtja, amikhez dimenziótáblák kapcsolódnak, akár csak egy csillagséma esetében. A séma lényegében több csillagséma összeépítése oly módon, hogy legyenek a dimenziótáblák között átfedések. Ezen kívül a séma részét képezheti még akár hópehely séma is, így ez egy nagyon sokoldalú sémának tekinthető.



4.3. ábra Egy galaxis séma ábrája

4.2.3. A hópehely séma

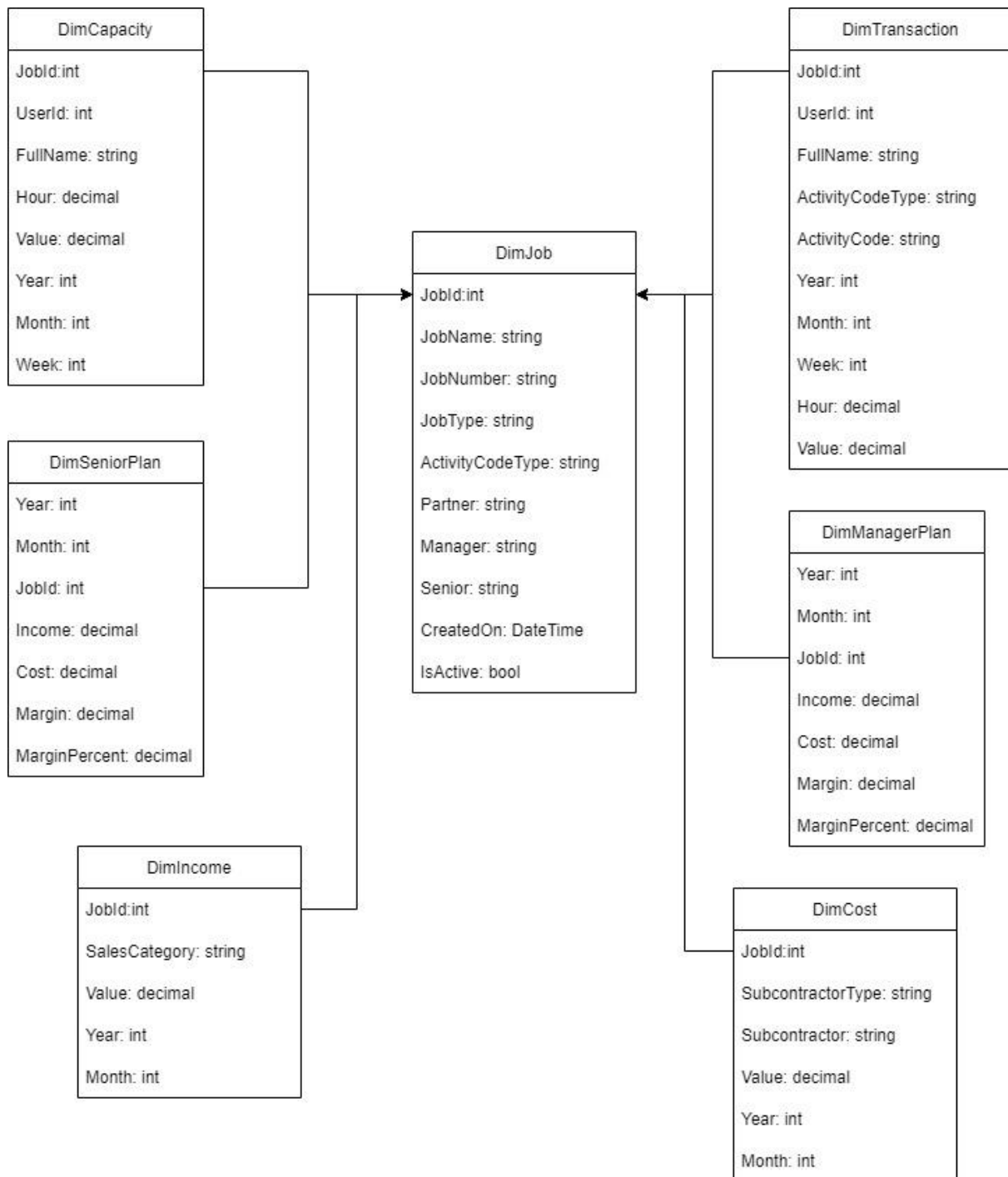
A harmadik séma, amit be fogok mutatni, az pedig a hópehely séma. Ez a séma is az alakjából kapta a nevét, ugyanis a dimenziótáblák minél mélyebbre mennek, annál inkább hasonlít egy hópehelyre a kialakított modell. Ez a séma egy olyan módon kialakított csillagséma, melynek dimenziótáblái normalizált módon tárolják az adatokat, ezzel csökkentve a redundancia mértékét. Ez viszont egyben az előnye és a hátránya is a rendszernek. Azzal, hogy csökkentettük a redundanciát, a rendszer lényegében kisebb lett, könnyebben olvashatóbb ezáltal, viszont ha minden adatra szükségünk van, ezek újabb kapcsolatokat (joinokat) kell a lekérdezésekbe beépítenünk, ezzel viszont lassíthatjuk azokat. [9]



4.4. ábra Egy hópehely séma ábrája

4.3. A rendszerem adattárháza

Az én adattárházam egy 7 táblából áll. Sok táblát kivettem a képből, illetve átalakítottam, hogy minél egyszerűbb, és kevesebb táblából álló adattárházat kapjak. A séma úgy lett kialakítva, hogy heti és havi szintre lett lebontva minden információ, így könnyedén lehet heti, havi, illetve éves értékeléseket csinálni.



4.5. ábra Az adattárházam felépítése

Tábláimat tekintve nem sok változás történt. Az alap táblák, mint a tranzakciók, bevételek, költségek, vagy épp a kapacitás, mind megmaradtak kicsit átalakítva, viszont bekerült egy újabb tábla DimPlanning néven, ami az AdhocTemplates és a Templates táblából kalkulált értékeket tárolja. Ezek a táblák elég nagyok lehetnek, így fontos, hogy az adattárházba már egy kikalkulált értéket tároljunk, számunkra ugyanis csak ez a fontos, az nem számít, miből alakultak ezek a számok. Ezek a adattárház sémám dimenziótáblái:

DimCost (tartalmazza a havi lebontott költségeket):

- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- Subcontractor: az alvállalkozó nevét tartalmazza (elsődleges kulcs)
- SubcontractorType: az alvállalkozó típusának nevét tartalmazza (elsődleges kulcs)
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Value: az adott hétre a költségek összegzett értéke

DimManagerPlan (tartalmazza a managerek havi tervezett adatait):

- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Income: a tervezett bevétel
- Cost: a tervezett költség
- Margin: a tervezett profit
- MarginPercent: a tervezett profit százalék

DimIncome (tartalmazza a havi lebontott bevételeket):

- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- SalesCategory: a bevétel típusa (elsődleges kulcs)
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Value: az adott hétre a bevételek összegzett értéke

DimCapacity (tartalmazza a hetekre lebontott személyekre allokált időt):

- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- Week: melyik hétre vonatkozik (elsődleges kulcs)
- UserId: a felhasználó azonosítója (elsődleges kulcs)
- FullName: a felhasználó neve
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Hour: hány óra volt allokálva
- Value: mennyi ennek az allokált időnek a becsült költsége

DimSeniorPlan (tartalmazza a seniorok havi tervezett adatait):

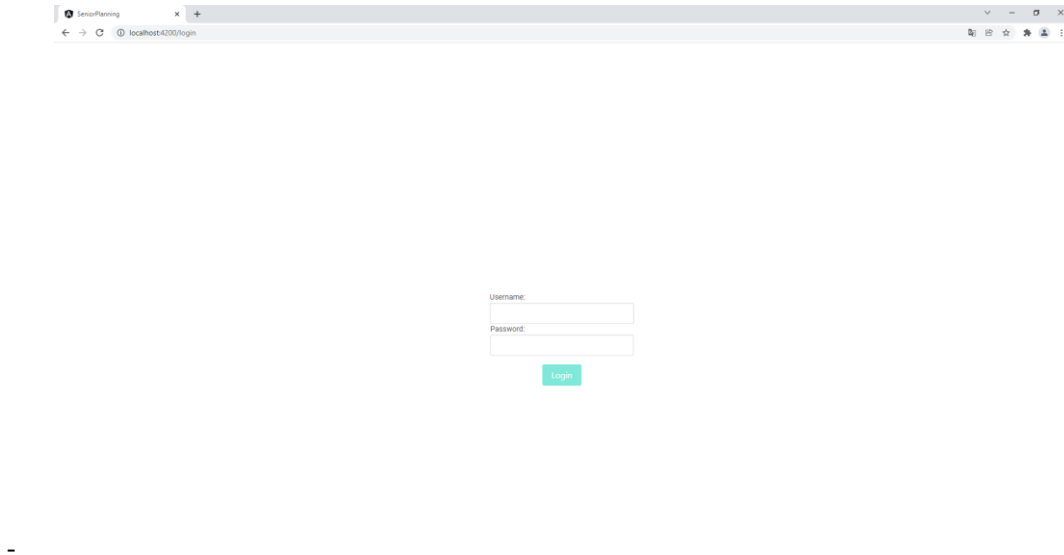
- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Income: a tervezett bevétel
- Cost: a tervezett költség
- Margin: a tervezett profit
- MarginPercent: a tervezett profit százalék

DimTransactions (tartalmazza a hetekre lebontott munkaórákat):

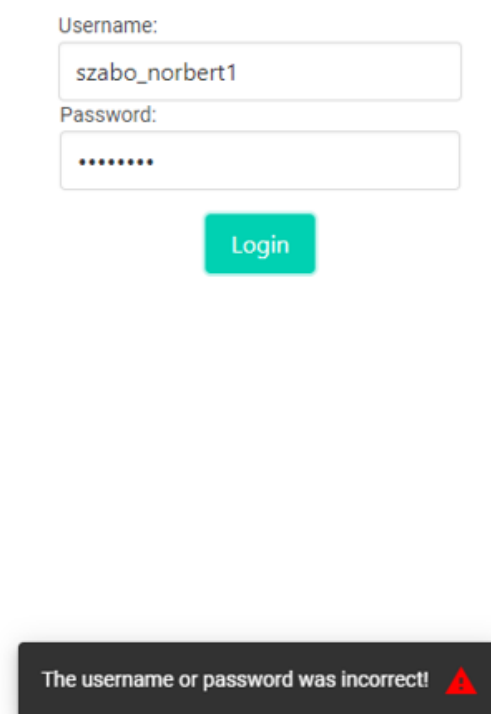
- Year: melyik évre vonatkozik (elsődleges kulcs)
- Month: melyik hónapra vonatkozik (elsődleges kulcs)
- Week: melyik hétre vonatkozik (elsődleges kulcs)
- UserId: a felhasználó azonosítója (elsődleges kulcs)
- FullName: a felhasználó neve
- ActivityCode: milyen tevékenységre terhelt időt a felhasználó (elsődleges kulcs)
- ActivityCodeType: milyen tevékenység típusra terhelt időt a felhasználó
- JobId: a job tábla azonosítója (elsődleges kulcs)
- Value: az adott hétre a bevételek összegzett értéke

5. Felhasználói dokumentáció

A rendszerben 4 fajta jogosultság osztható ki, és ezek alapján lesznek láthatóak az egyes oldalak. A legelső oldal, amivel találkozunk, az nem más, mint a bejelentkező felület. A bejelentkezés egy egyszerű felhasználónév/jelszó párossal történik, ami után vagy beléptet a rendszer, vagy pedig hibaüzenetet dob nem megfelelő belépési adatok esetén.

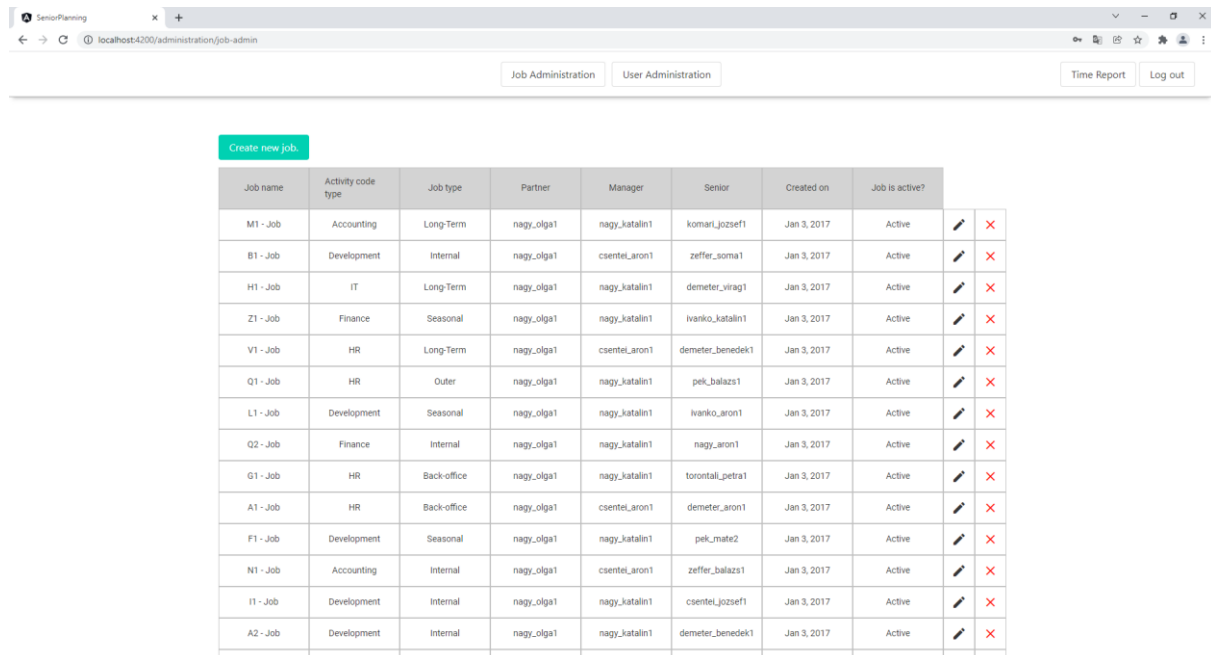


5.1. ábra Az a bejelentkező képernyő



5.2. ábra Hibás bejelentkezés példája

Bejelentkezés után 4 fajta képernyővel találkozhatunk. Ez attól függ, hogy mely felhasználóval léptünk be. Sorban be is mutatom őket. Kezdjük az adminisztrátorral.

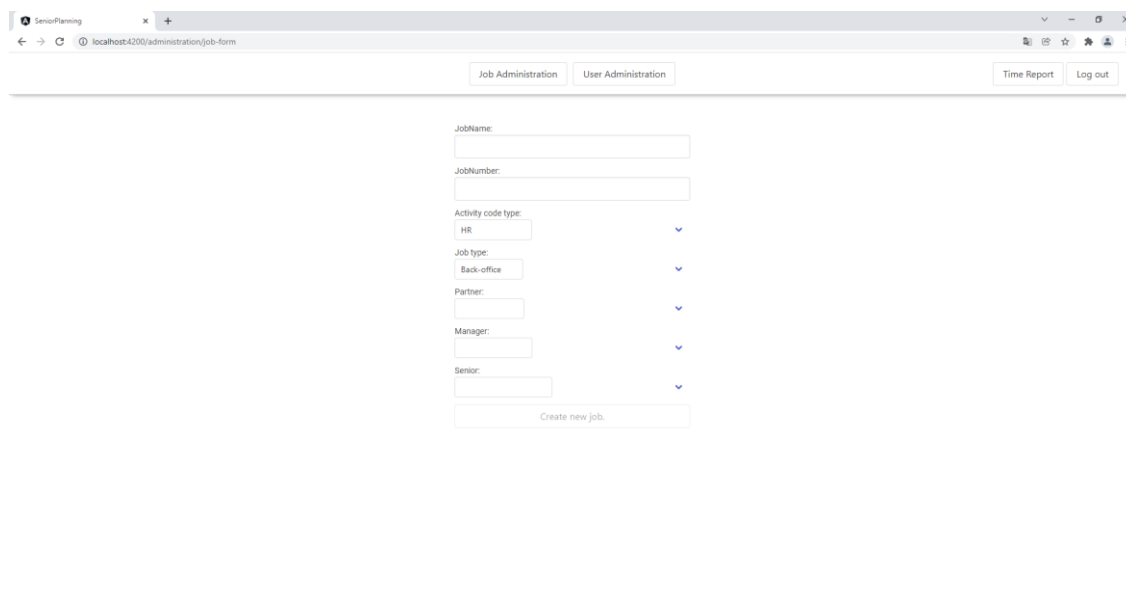


Job name	Activity code type	Job type	Partner	Manager	Senior	Created on	Job is active?
M1 - Job	Accounting	Long-Term	nagy_olga1	nagy_katalin1	komari_jozsef1	Jan 3, 2017	Active
B1 - Job	Development	Internal	nagy_olga1	csentei_aron1	zeffer_soma1	Jan 3, 2017	Active
H1 - Job	IT	Long-Term	nagy_olga1	nagy_katalin1	demeter_vilag1	Jan 3, 2017	Active
Z1 - Job	Finance	Seasonal	nagy_olga1	nagy_katalin1	ivanko_katalin1	Jan 3, 2017	Active
V1 - Job	HR	Long-Term	nagy_olga1	csentei_aron1	demeter_benedek1	Jan 3, 2017	Active
Q1 - Job	HR	Outer	nagy_olga1	nagy_katalin1	pek_balazs1	Jan 3, 2017	Active
L1 - Job	Development	Seasonal	nagy_olga1	nagy_katalin1	ivanko_aron1	Jan 3, 2017	Active
Q2 - Job	Finance	Internal	nagy_olga1	nagy_katalin1	nagy_aron1	Jan 3, 2017	Active
G1 - Job	HR	Back-office	nagy_olga1	nagy_katalin1	torontali_petra1	Jan 3, 2017	Active
A1 - Job	HR	Back-office	nagy_olga1	csentei_aron1	demeter_aron1	Jan 3, 2017	Active
F1 - Job	Development	Seasonal	nagy_olga1	nagy_katalin1	pek_mate2	Jan 3, 2017	Active
N1 - Job	Accounting	Internal	nagy_olga1	csentei_aron1	zeffer_balazs1	Jan 3, 2017	Active
I1 - Job	Development	Internal	nagy_olga1	nagy_katalin1	csentei_jozsef1	Jan 3, 2017	Active
A2 - Job	Development	Internal	nagy_olga1	nagy_katalin1	demeter_benedek1	Jan 3, 2017	Active

5.3. ábra Adminisztrátori felület dashboardja

Adminisztrátorként belépve ez az oldal fogad minket. Itt látható az összes munka (job) a rendszerben. Lehet deaktiválni őket, avagy szerkeszteni, avagy újat létrehozni. Felül látható, hogy ide a „Job Administration” menüikonra kattintva juthatunk, míg ha a felhasználókat szeretnénk látni, azt a „User Administration” oldalon tehetjük meg.

A felhasználómódosítás esetén az alábbi oldalt fogjuk látni.



JobName:

JobNumber:

Activity code type:

Job type:

Partner:

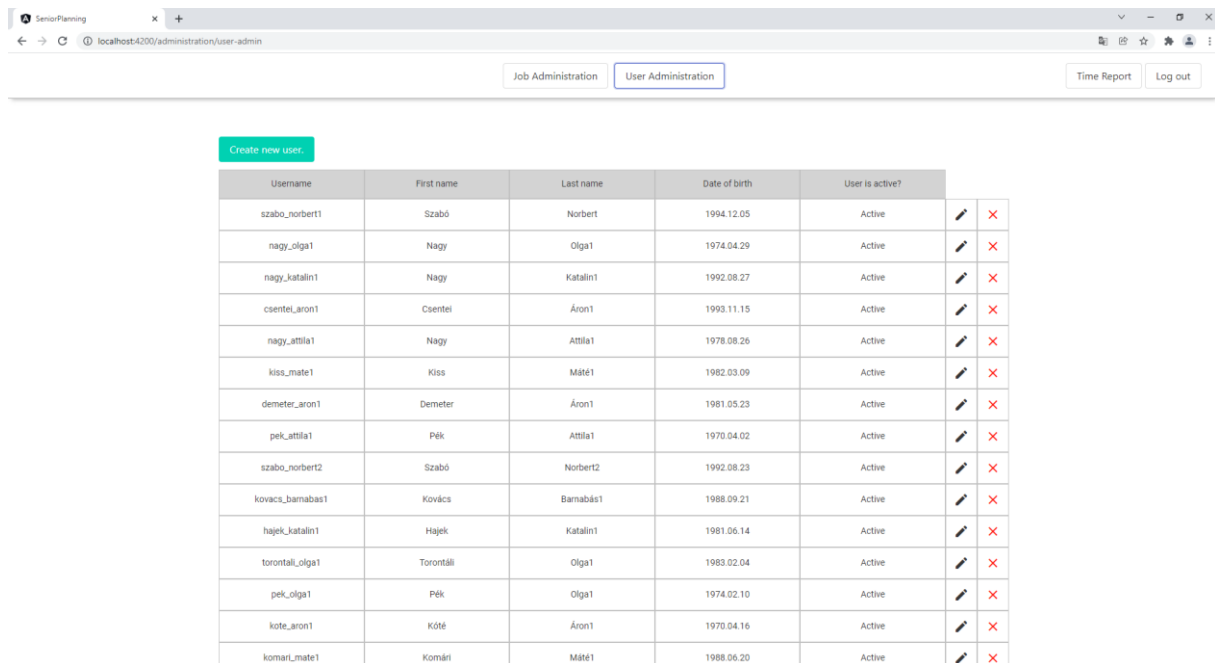
Manager:

Senior:

5.4. ábra Job adminisztrációs form

Ide betölti az adatokat, hogyha az előző felületről szeretnénk egy felhasználót módosítani. Ha a „Create new job” gombra kattintunk, akkor pedig egy üres formot kapunk.

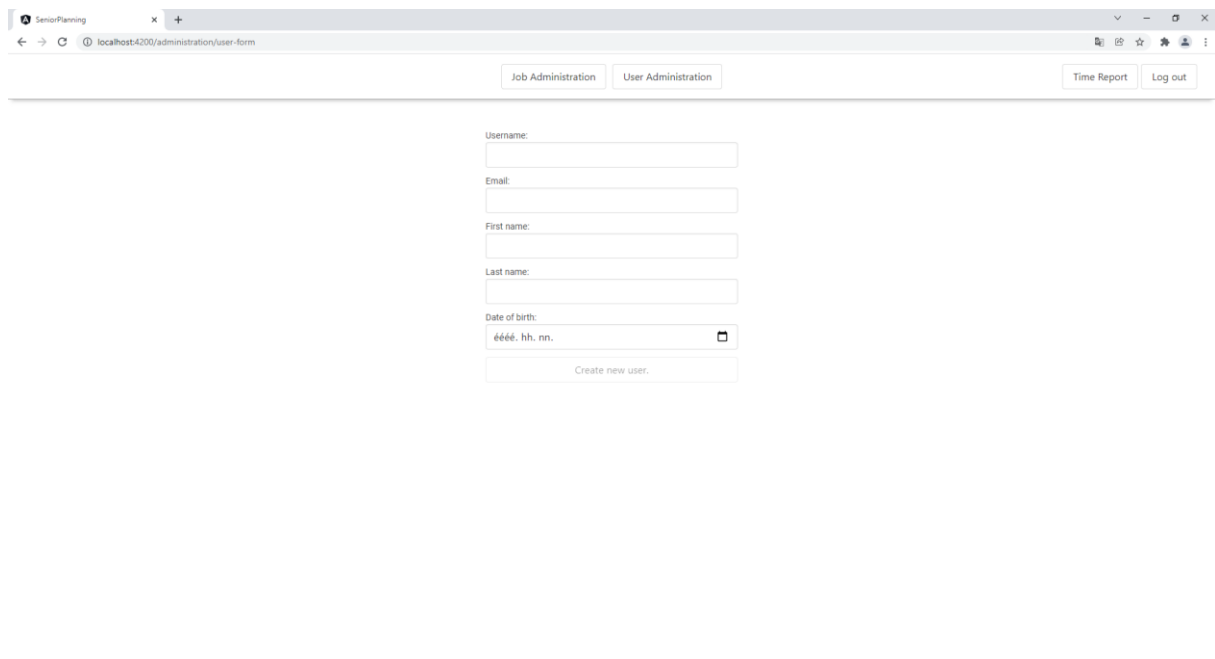
Az adminisztrátor következő oldala a felhasználói oldal, ahol hasonlóan az előzőhöz, inaktíválhatunk usert, esetleg felvihetünk újat, vagy meglévőket módosíthatunk.



Username	First name	Last name	Date of birth	User is active?		
szabo_norbert1	Szabó	Norbert	1994.12.05	Active		
nagy_olga1	Nagy	Olga1	1974.04.29	Active		
nagy_katalin1	Nagy	Katalin1	1992.08.27	Active		
csentei_aron1	Csentei	Áron1	1993.11.15	Active		
nagy_attila1	Nagy	Attila1	1978.08.26	Active		
kiss_mate1	Kiss	Máté1	1982.03.09	Active		
demeter_aron1	Demeter	Áron1	1981.05.23	Active		
pek_attila1	Pék	Attila1	1970.04.02	Active		
szabo_norbert2	Szabó	Norbert2	1992.08.23	Active		
kovacs_barnabas1	Kovács	Barnabás1	1988.09.21	Active		
hajak_katalin1	Hajak	Katalin1	1981.06.14	Active		
torontali_olga1	Torontáli	Olga1	1983.02.04	Active		
pek_olga1	Pék	Olga1	1974.02.10	Active		
kote_aron1	Köté	Áron1	1970.04.16	Active		
komarim_mate1	Komári	Máté1	1988.06.20	Active		

5.5. ábra User adminisztrációs felület

A felhasználó módosító form hasonlóképpen működik, mint a jobok esetében. Betölti, ha már meglévőt módosítanánk, ellenkező esetben új formot hoz létre.



Username:

Email:

First name:

Last name:

Date of birth:

5.6. ábra User adminisztrációs form

A következő bemutatott felhasználói felülete a senioré lesz. Bejelentkezés után a senior dashboard felület fog megjelenni a felhasználó számára.

Senior Dashboard	Capacity Planning	Senior Job Planning	Reporting	Time Report	Log out
------------------	-------------------	---------------------	-----------	-------------	---------

Senior dashboard

From: 2021. december To: 2022. február

Weekly grouped capacity values

Job name	2021 49. week	2021 50. week	2021 51. week	2021 52. week	2022 1. week	2022 2. week	2022 3. week	2022 4. week
Q3 - Job	184500	184500	184500	184500	184500	184500	184500	184500
R2 - Job	215600	241600	167600	235600	309600	247600	271600	412100
C1 - Job	213000	68000	0	202000	94000	34000	154000	138500

Weekly grouped transaction values

Job name	2021 49. week	2021 50. week	2021 51. week	2021 52. week	2022 1. week	2022 2. week	2022 3. week	2022 4. week
Q3 - Job	0	0	0	0	0	0	0	0
R2 - Job	0	0	0	0	0	0	0	0
C1 - Job	0	0	0	0	0	0	0	0

5.7. ábra Senior dashboard felülete

Ezen a felületen a senior a saját dashboard adatait láthatja időszakosan, heti szintre lebontva. Az első táblázat a senior által felvitt capacity tervek tartalmazza, míg alatta a valós időt láthatjuk, amit ténylegesen dolgoztak az asszisztensei. Fenti menüpontokból a következő a „Capacity Planning” lesz.

SeniorPlanning	Senior Dashboard	Capacity Planning	Senior Job Planning	Reporting	Time Report	Log out
----------------	------------------	-------------------	---------------------	-----------	-------------	---------

Job name: Q3 - Job

Name	2021/50	2021/51	2021/52	2022/1	2022/2	2022/3	2022/4	2022/5	2022/6	2022/7	2022/8	2022/9	2022/10
Szabó Norbert2	40	40	40	40	40	40	40	40	40	40	40	40	0
Kovács Barnabás1	5	5	5	5	5	5	5	5	5	5	5	5	0
Hajek Katalin1	30	30	30	30	30	30	30	30	30	30	30	30	0
Torontáli Olga1	20	20	20	20	20	20	20	20	20	20	20	20	0
Faragó Attila1	10	10	10	10	10	10	10	10	10	10	10	10	0
Csentei Attila1	0	0	0	0	0	0	0	0	0	0	0	0	40

Save capacities

5.8. ábra Capacity Planning tervező felülete

A senior itt látja az általa betervezhető felhasználókat és munkákat heti szintre lebontva. Itt megadhatja, hogy egy héten egy adott felhasználónak mely munkán mennyi időt kell legalább dolgoznia. Ha egy asszisztens több seniorhoz is tartozik, akkor az ideje megoszlik kettejüknel.

„Senior Job Planning” menüpont a következő.

Senior Dashboard
Capacity Planning
Senior Job Planning
Reporting
Time Report
Log out

Senior job planning

From: 2021. december
Job name: Q9 - Job

	2021/12	2022/1	2022/2	2022/3	2022/4	2022/5	2022/6	2022/7	2022/8	2022/9	2022/10	2022/11
Income	450	400	400	400	400	400	400	400	400	400	400	400
Cost	150	150	150	150	150	150	150	150	150	150	150	150
Margin	200	200	200	200	200	200	200	200	200	200	200	200
Margin Percent	80	80	80	80	80	80	80	80	80	80	80	80

Save senior plans

5.9. ábra Senior Job Planning tervező felülete

Itt a senior által előre betervezett bevételt, költséget, margint és margin százalékot adhatja meg egyes jobokra, havi lebontásban. Ezekre a tervszámokra azért van szükség, mert később a tényadatokkal összehasonlítva meg tudjuk vizsgálni, hogy a tervezett összegek mennyire voltak a valóságnak megfelelőek.

Az utolsó a „Reporting” fül lesz, de mivel az osztozik a Manager, Senior és Partner jogosultság között, azt a végén mutatom be részletesen.

Nézzük a következő jogosultságot, mégpedig a manager felhasználóit. A managerek bejelentkezés után egyből a dashboardjukra kerülnek.

Manager Dashboard
Manager Job Planning
Reporting
Time Report
Log out

Manager dashboard

From: 2021. december
To: 2022. november
Job filter:
Senior filter:

Job name	Type/Senior name	2021/12	2022/1	2022/2	2022/3	2022/4	2022/5	2022/6
▼ M1 - Job	Komár József1							
	Capacity	48000	138000	128000	72000	0	0	0
	Transaction	0	0	0	0	0	0	0
	FactIncome	0	0	0	0	0	0	0
	FactCost	0	0	0	0	0	0	0
	Senior Plan Cost	0	0	0	0	0	0	0
	Senior Plan Income	0	0	0	0	0	0	0
	Senior Plan Margin	0	0	0	0	0	0	0
	Senior Plan Margin %	0	0	0	0	0	0	0
▼ H1 - Job	Demeter Virág1							
	Capacity	302000	48000	128000	48000	0	0	0
	Transaction	37600	0	0	0	0	0	0
	FactIncome	0	0	0	0	0	0	0
	FactCost	0	0	0	0	0	0	0
	Senior Plan Cost	0	0	0	0	0	0	0
	Senior Plan Income	0	0	0	0	0	0	0
	Senior Plan Margin	0	0	0	0	0	0	0
	Senior Plan Margin %	0	0	0	0	0	0	0
▼ Z1 - Job	Ivancs Katalin1							
	Capacity	150000	120000	95000	142000	0	0	0

5.10. ábra Manager dashboard felülete

Ezen az oldalon a manager láthatja egyes jobokra lebontva, havi lebontásban az adatokat. Az itt látott adatok a bevételek, költségek, a munkára allokalált idő költsége, és a senior által

betervezett számadatok. A könnyebb átláthatóság érdekében lehet szűrni jobokra, seniorokra, avagy az időszak is megadható, hogy mettől-meddig mutasson adatokat havi bontásban.

A következő és egyben utolsó manager-specifikus oldal az a Manager Job Planning.

	2021/12	2022/1	2022/2	2022/3	2022/4	2022/5	2022/6	2022/7	2022/8	2022/9	2022/10	2022/11
Income	300	300	300	300	300	300	300	300	300	300	300	300
Cost	1500	1500	1500	1500	1500	1500	1500	1500	1500	1500	1500	1500
Margin	20	20	20	20	20	20	20	20	20	20	20	20
Margin Percent	30	0	0	40	0	0	0	0	0	0	0	0

5.11. ábra Manager Job Planning tervező felülete

Itt a manager is megadhat tervszámokat az adott munkáihoz hasonló felépítésben, mint a senior. A lényege az egésznek, hogy így 2 tervszám is összehasonlítható, hogy a manager, illetve a senior, aki más szemszögből közelítette meg a terveket és a munkát magát, mennyire tudta jól behatárolni a munka lehetséges költségeit.

Az utolsó fontosabb jogosultság a partneri jogosultság, ami a cégvezetőké lényegében. Ők nem terveznek, ők csupán egy dashboardot látnak, amiken a teljes céges adatokat látják egyes munkákhoz, így azonnali rálátást kapnak akármelyik számra egyes munkákkal kapcsolatban.

Partner Dashboard

Time Report Log out

Manager dashboard

From: 2021. január To: 2021. december Job filter: Manager filter:

Job name	Type	Manager name	2021/1	2021/2	2021/3	2021/4	2021/5	2021/6	2021/7
▼ M1 - Job		Nagy Kata							
	Capacity		433500	308000	236000	324000	48000	128000	208000
	Transaction		3987650	2632300	2331000	2319950	1692900	2693650	2085650
	FactIncome		4638	0	4162	4240	2899	2796	2235
	FactCost		3268	0	5879	4911	3140	4593	5339
	Senior Plan Cost		14200	14200	14200	14200	14200	14200	14200
	Senior Plan Income		6600	6600	6600	6600	6600	6600	6600
	Senior Plan Margin		-7600	-7600	-7600	-7600	-7600	-7600	-7600
	Senior Plan Margin %		46.48	46.48	46.48	46.48	46.48	46.48	46.48
	Manager Plan Cost		3500	3500	3500	3500	3500	3500	3500
	Manager Plan Income		14300	14300	14300	14300	14300	14300	14300
	Manager Plan Margin		10800	10800	10800	10800	10800	10800	10800
	Manager Plan Margin %		408.57	408.57	408.57	408.57	408.57	408.57	408.57
▼ B1 - Job		Csentei Áron							
	Capacity		48000	176000	48000	94000	36000	184000	224000
	Transaction		1427000	1688950	1726000	2222700	2065000	2050950	1446400
	FactIncome		8432	0	5990	3565	3828	3445	4919
	FactCost		4591	0	2208	3719	1872	5545	4569
	Senior Plan Cost		10400	10400	10400	10400	10400	10400	10400
	Senior Plan Income		6200	6200	6200	6200	6200	6200	6200

5.12. ábra Partner dashboard felülete

Ezek után már csak 2 oldal maradt. Az első legyen a Reporting, amit managerek, seniorok és partnerek is elérhetnek.

Senior Dashboard Capacity Planning Senior Job Planning Reporting Time Report Log out

Reporting

From: 2021. december To: 2022. november

Select a report: Full senior report

Job filter:

Generate & Download Report

5.13. ábra Reporting felület

A reporting célja, hogy szűrhetően (ha szeretné a felhasználó) excel exportokat tudjon generálni. Minden jogosultságnak saját reportjai vannak, így biztosítva, hogy mindenki csak azokat a számokat láthatja, amire szüksége is lehet.

Utolsó oldal az a „Time Report” oldala lesz.

The screenshot shows a web browser window with the URL `localhost:4200/time-report`. The page has a header with a "Partner Dashboard" button and "Time Report" and "Log out" links. The main content area is titled "Time Report" and includes a date selector set to "2021. 12. 15.". Below this is a table with four columns: "Job name", "Activity code", "Description", and "Hour". The table contains two rows: one for "Z1 - Job" with "Client-specific" activity and 4 hours, and another for "M1 - Job" with "Management w..." activity and 3 hours. A green "Save Time Report changes" button is at the bottom, along with a small green plus icon.

Job name	Activity code	Description	Hour
Z1 - Job	Client-specific	Client-specific	4
M1 - Job	Management w...	Management meeting	3

5.14. ábra Time Report felület

Az oldal azt a célt szolgálja, hogy mindenki fel tudja részletesen vinni, hogy egy adott munkán aznap pontosan mennyi időt foglalkozott. Ezt az adatot pedig a tervadatokkal később összehasonlítjuk, és megállapítható, hogy a tervezett adatok mennyire voltak valósak a tényadatokkal.

Tesztelési jegyzőkönyv

A teljes rendszer tesztelését manuálisan végeztem, minden funkciót backend és frontend oldalon is teszteltem. Ezen kívül a jogosultságot teszteltem programszinten is integration tesztekkel, amikről a kép itt is van.

```
✓ SeniorPlanningTest.IntegrationTests (6 tests) Success
  ✓ SeniorPlanning.IntegrationTests (6 tests) Success
    ✓ AuthenticationControllerTests (6 tests) Success
      ✓ CheckManagerOnlyPolicy_WithInvalidPermissions_ReturnsStatus403 Success
      ✓ CheckManagerOnlyPolicy_WithValidPermissions_ReturnsStatus200 Success
      ✓ CheckSeniorOnlyPolicy_WithInvalidPermissions_ReturnsStatus403 Success
      ✓ CheckSeniorOnlyPolicy_WithValidPermissions_ReturnsStatus200 Success
      ✓ Login_WithNotValidUserName_ReturnsBadRequestAndNotFoumdUserNameErrorCode Success
      ✓ Login_WithValidLoginParameters_ReturnsStatus200AndRefreshToken Success
```

5.15. ábra Rendszer integrációs tesztjei

6. Szakdolgozat összegzése

A szakdolgozatomban bemutattam, hogy szeretnék lefejleszteni egy tervező és riport készítő rendszert. Ehhez végeztem egy piackutatást, ahol utánajártam, hogy a piac milyen rendszereket kínál, s azok milyen hasonlóságot, avagy különbséget mutatnak az én rendszeremhez képest. Szemléltettem, hogy milyen platformokat és szoftvereket használhatnék, és miért azok mellett döntöttem. Ismertettem a rendszerem terveit, bemutattam annak felépítését. Elkészítettem hozzá különféle diagramokat a könnyebb átláthatóság érdekében, illetve egy adatbázistervet is készítettem már, amiből könnyen látható, hogy milyen adatokkal dolgozna a rendszer. Ezen felül megterveztem annak adattárházát is.

Továbbfejlesztési javaslatként a jövőben egy sokkal felhasználóbarátabb UI felület szükséges volna, ami nem csak színekkel segítené a felhasználót, hanem elrendezés és átláthatósági szempontból is. Továbbá a teljes rendszert 100%-os unit és integration tesztekkel kellene lefedni.

7. Summary

My priority was to show you that reporting and planning system, what I had in mind. First, I needed some market research, so I can have a better picture of what the current market can actually offer right now. This way I could determine the differences and similarities between my system and theirs. I wrote down what kind of platform would I use, and why I chose them. I created diagrams, database schemas (both OLAP and OLTP) and the system design, so the system can be understood much easily as a whole.

As a future plan, the system UI should be refactored, because right now it's not really user-friendly. It should have a much more colorful and user-friendly layout, where the user can easily determine the different numbers and values.

8. Irodalomjegyzék

- [1] SAP Business One: (<https://www.sap.com/hungary>), utoljára megtekintve: 2020. 11. 15.
- [2] Vectory: (<https://www.vector.hu/vectory-vallalatiranyitasi-rendszer>), utoljára megtekintve: 2020. 11. 15.
- [3] Revolution – deep.erp: (<https://www.revolution.hu/deep/deep-erp>), utoljára megtekintve: 2020. 11. 15.
- [4] Online tranzakciófeldolgozás (OLTP): (<https://docs.microsoft.com/hu-hu/azure/architecture/data-guide/relational-data/online-transaction-processing>), utoljára megtekintve: 2020. 11. 15.
- [5] Online analitikus feldolgozás (OLAP): (<https://docs.microsoft.com/hu-hu/azure/architecture/data-guide/relational-data/online-analytical-processing>), utoljára megtekintve: 2020. 11. 15.
- [6] A csillagséma áttekintése: (<https://docs.microsoft.com/hu-hu/power-bi/guidance/star-schema>), utoljára megtekintve: 2020. 11. 15.
- [7] Star schema: (https://en.wikipedia.org/wiki/Star_schema), utoljára megtekintve: 2020. 11. 15.
- [8] Galaxy Schema: (<https://www.educba.com/galaxy-schema/>), utoljára megtekintve: 2020. 11. 15.
- [9] Snowflake schema: (https://en.wikipedia.org/wiki/Snowflake_schema), utoljára megtekintve: 2020. 11. 15.

9. Ábrajegyzék

2.1. ábra az S/4 HANA kezelőfelete	10
2.2. ábra deep.erp kezelőfelülete	13
3.1. ábra A rendszerem használati eset diagramja.....	17
3.2. ábra A szeniori szerepkör szekvencia diagramja.....	18
4.1. ábra A rendszerem OLTP adatbázisának szerkezete	20
4.2. ábra Egy csillag séma ábrája	25
4.3. ábra Egy galaxis séma ábrája	26
4.4. ábra Egy hópehely séma ábrája	26
4.5. ábra Az adattárházam felépítése	27
5.1. ábra Az a bejelentkező képernyő	30
5.2. ábra Hibás bejelentkezés példája.....	30
5.3. ábra Adminisztrátori felület dashboardja	31
5.4. ábra Job adminisztrációs form.....	31
5.5. ábra User adminisztrációs felület	32
5.6. ábra User adminisztrációs form.....	32
5.7. ábra Senior dashboard felülete	33
5.8. ábra Capacity Planning tervező felülete	33
5.9. ábra Senior Job Planning tervező felülete	34
5.10. ábra Manager dashboard felülete.....	34
5.11. ábra Manager Job Planning tervező felülete.....	35
5.12. ábra Partner dashboard felülete	36
5.13. ábra Reporting felület.....	36
5.14. ábra Time Report felület.....	37
5.15. ábra Rendszer integrációs tesztjei	37