

DIPLOMAMUNKA

Hidi Viktória

Debrecen

2010

Debreceni Egyetem

Informatikai Kar

Autoimmun betegségek adatbázisának relációs alapokra helyezése

Témavezető:

Dr. Végh János

Prof. Egyetemi tanár

Külső konzulens:

Vincze Melinda

szakértő, munkatárs Klinikai Immunológia Tanszék

Hidi Viktória

Programtervező Matematikus

Debrecen, 2010

1 TARTALOMJEGYZÉK

1	TARTALOMJEGYZÉK	3
2	BEVEZETÉS	4
3	ADAT - adatbázis - INFORMÁCIÓ	5
3.1	Adat és Információ.....	6
3.2	Adatbázis (DB) és Adatbázis kezelő rendszer (DBMS).....	6
4	ADATGYŰJTÉS és ELEMZÉS	8
4.1	Fogalmak tisztázása	8
4.2	Interjúk készítése	10
4.3	A „hagyaték” elemzése.....	12
5	TERVEZÉS	20
6	MODELLEZÉS	22
6.1	Az ER modell (FOGALMI szint).....	23
6.2	A Relációs modell (LOGIKAI szint).....	26
6.3	A relációsémák normalizációja.....	31
6.4	Megvalósítás (FIZIKAI szint)	33
7	ADATMIGRÁCIÓ	35
7.1	Extraktálás	35
7.2	Transzformálás.....	36
7.3	Betöltés	36
8	LEKÉRDEZÉSEK, vagyis az EREDMÉNY	38
8.1	Kérdés 1.	39
8.2	Kérdés 2.	40
8.3	Kérdés 3.	42
8.4	Kérdés 4.	43
8.5	Alkalmazott szoftverek.....	45
9	IRODALOMJEGYZÉK	51
9.1	Könyvek.....	51
9.2	Online irodalom	51
9.3	Szoftverek	51
10	ÖSSZEGZÉS	52
10.1	Köszönetnyilvánítás.....	53
11	FÜGGELÉK	54
11.1	Az autoimmun betegségek és a Myositis.	54
11.2	ER modell	56
11.3	Ábrajegyzék	58
11.4	Etikai nyilatkozat	60

2 BEVEZETÉS

A témaválasztás oka.

Diplomamunkám témájának kiválasztása előtt nagy dilemmában voltam, mivel az informatikai kar által hirdetett kiírások közül több is érdekelt. Főleg az adatbázisokkal kapcsolatos témaköröket találtam izgalmasnak. A döntést jelenlegi témavezetőm, Dr. Végh János tanácsára hoztam meg, aki egy olyan témát ajánlott figyelmembe, ami azonnal felkeltette az érdeklődésemet. A DEOEC III. számú Belgyógyászati Klinika professzor asszonya Végh tanár urat kérte fel, keressen számukra egy informatikus hallgatót, aki megtervezné, illetve helyesen kialakítaná az intézet adatbázisát, továbbá elkészítené az adatbázishoz kapcsolódó alkalmazást. A munka érdekesnek ígérkezett, és nagy kihívás volt számomra. Úgy döntöttem, hogy belevágok. Akkor (és jelenleg is) egy orvos informatikai cégnél dolgoztam, a téma pedig szintén egy orvosi adatbázis, később pedig akár egy hozzá kapcsolódó felület elkészítése is kitűzött cél lehetne. A véletlen műve lenne? Hiszem, hogy nem. Semmi sem a véletlen műve. „Az Úr vezet majd szüntelen” /Ézsaiás 58,11/

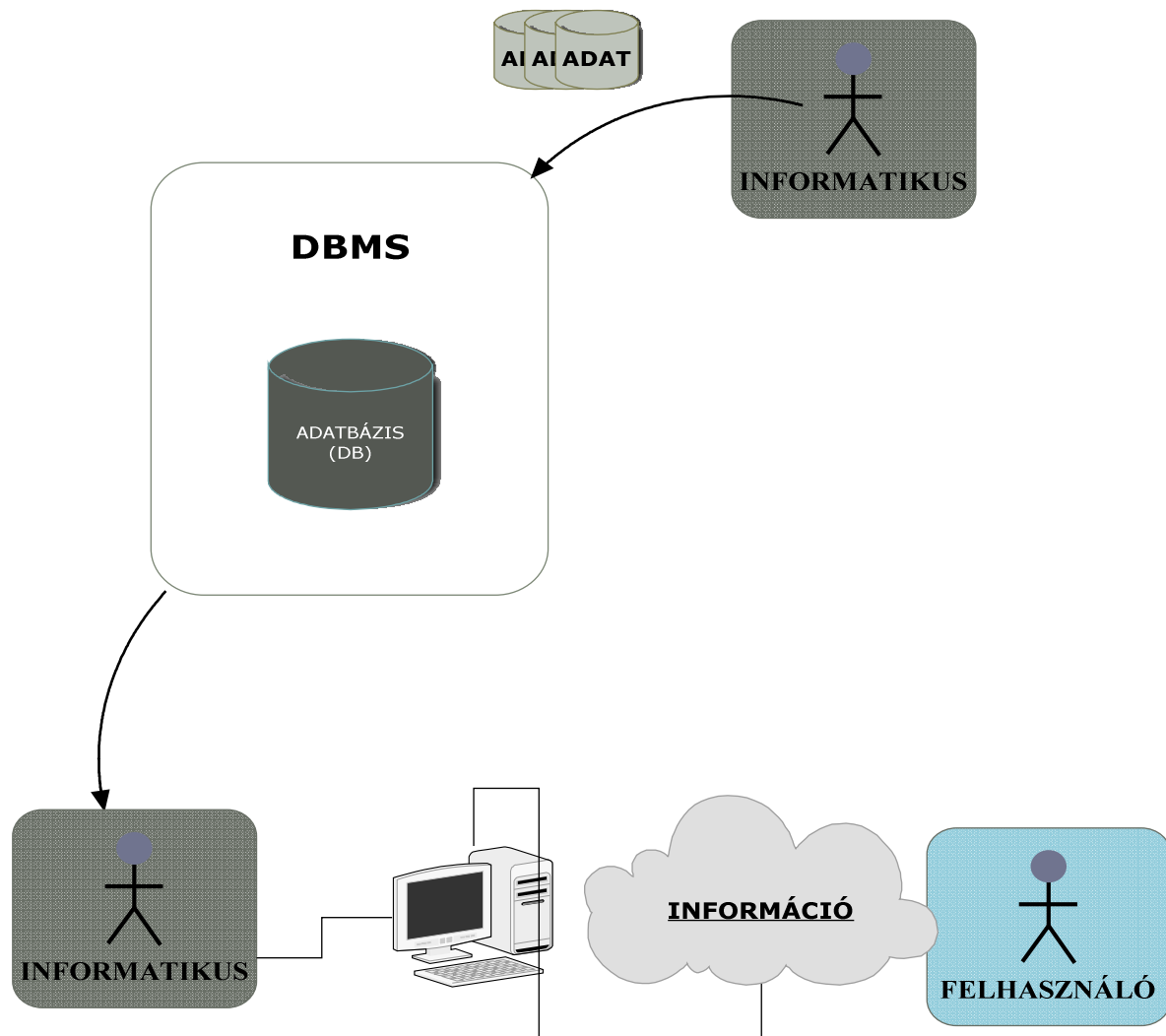
A felkérés körülményei.

A Debreceni Egyetem Orvos-és Egészségtudományi Centrum III. számú Belgyógyászati Klinikáján 1985 óta gondoznak myositisben szenvedő betegeket. Dr. Dankó Katalin indította el a szakrendelést és jelenleg is az ő vezetésével működik. Az elmúlt években több mint 400 betegnél diagnosztizáltak myositist, akik közül 200-250 páciens rendszeresen visszajár a szakrendelésre, kezdetben havonta, később a tünetek, panaszok alakulásától függően ritkábban. Mivel ritka betegségcsoportról van szó és a kialakulásának, illetve lefolyásának részletei nem teljesen ismertek, Európában és a világon is számos orvosi, biológiai, genetikai kutatás témái az idiopáthiás inflammatorikus myopáthiák. A DEOEC III. számú Belgyógyászati Klinika több projektben is részt vesz, szoros együttműködésben svéd, svájci, izraeli és angol kutatócsoportokkal. Ezen kívül létrehoztak Magyarországon egy jól működő, orvosokat, gyógytornászokat, molekuláris biológusokat, patológusokat tömörítő szervezetet, az Országos Myositis Munkacsoportot, melynek célja, hogy a myositist korai stádiumban felismerjék és a megfelelő szakemberhez irányítsák a betegeket.

A Myositisről és az autoimmun betegségekről bővebben a függelékben található fejezetből lehet tájékozódni [[Myositis](#)].

3 ADAT - adatbázis - INFORMÁCIÓ

Ahhoz, hogy az 1. ábra szereplőinek kapcsolatát megérthessük, először is tisztázni kell az adat – információ – adatbázis (DB) – adatbázis kezelő rendszer (DBMS) – fogalmakat, ezek kapcsolatát, továbbá meg kell értenünk, hogyan kapcsolódik ezekhez az informatikus szakember.



1. ábra: ADAT - adatbázis - INFORMÁCIÓ

3.1 Adat és Információ

Az **adat** értelmezhető (észlelhető, érzékelhető, felfogható, megérthető), de nem értelmezett ismeret. [2]

„Ha az izom biopszia eredménye pozitív, akkor még el kell végezni a HRCT-t ahhoz, hogy felállíthassuk a diagnózist”. A mondat nem mindenki számára érthető, annak ellenére, hogy előttünk van írott formában és olyan nyelven, melyen tudunk olvasni, vagyis elvileg értelmezhető. Mégis csak annak mond valamit, aki tudja mi az, hogy izom biopszia. Az értelmezés azt jelenti, hogy a fogalmakat az általunk már ismertekhez kapcsoljuk. Ezért igaz az, hogy a mondatban lévő ismeret egyesek (a hozzáértők) számára adatot hordoz, mások számára semmit sem jelent (még adatot sem).

Az **információ** új ismeretté értelmezett adat [2], az a jelentés, amelyet az ember tulajdonít az adatnak.[6]

Az adat tényt közöl, tehát objektív, az ismeret személyesen értelmezett dolog, vagyis szubjektív. Tehát egy adatból nagyon sok információ születhet, mivel egyik ember más információvá értelmezheti ugyanazt az adatot, mint a másik.

3.2 Adatbázis (DB) és Adatbázis kezelő rendszer (DBMS)

Adatszerű ismeretről akkor beszélünk, ha az adatok adattételekben (mezők, rovatok) elrendezettek. Az **adatbázis (Database, DB)** ilyen adatszerűen tárolt és kezelt ismeretek szervezett együttese.[3] Az adatbázisok célja adatok megbízható, hosszútávon tartós (perzisztens) tárolása, és viszonylag gyors visszakereshetőségének biztosítása.

Legfontosabb két vonása az, hogy tudatosan szervezett és közösen használt ismeret együttes. A tudatos szervezés azt jelenti, hogy a valós jelenségeknek megfelelően egyre mélyebb ismeretek szerint rendezzük el benne az adatokat a saját igényeink szem előtt tartásával. Ezáltal létrejön egy azonos minőségű és jellemzőjű strukturált adathalmaz, melyet egy tárolására, lekérdezésére és szerkesztésére alkalmas szoftvereszközzel kezelünk.

Az **adatbázis-kezelő rendszer** (Database Management System, DBMS) több felhasználós, hálózatos környezetben adatbázisokhoz való hozzáférést, rendszeres és a felhasználói folyamatok zavartalan működését biztosító szoftveralkalmazás. [11]

Tehát az adat, amit tárolunk, az adatbázis, amiben tároljuk. Az adat információvá válik a feldolgozás révén, de csak ha értelmes módon nyerjük ki az adatbázisból. Ehhez az adatbázis szerkezetének olyannak kell lennie, hogy ki lehessen nyerni belőle ezeket. Az sem mindegy hogyan nyerjük ki. Mivel napjainkban szelektív, emberre szabott adatszolgáltatásra van szükség, az adatokat úgy kell onnan „előhalászni”, hogy embertársaink információt nyerhessenek belőle és azt úgy kell a megcélzott felhasználó elé tárni, hogy az könnyen értelmezhető legyen számára. Figyelni kell tehát arra is, hogy olyan ismeretet adjunk át, amelyet a felhasználó könnyen és gyorsan azzá az információvá tud értelmezni, amelyre éppen szüksége van, vagyis a gyakorlati alkalmazhatóságára. Itt jön képbe az informatikus (tervező), aki számára az adat jó vagy rossz tartalmú, jól vagy rosszul elrendezett jelsor. Viszont az információt, vagyis azt a többlettudást, amit a jó tartalmú és jól elrendezett jelsorokból nyerhet a felhasználó, ha közvetve is (pl. programok által) ő szolgáltatja.

Az első hazai orvostudományi szakfolyóirat megjelenésétől napjaink internetes könyvtári tájékoztató szolgáltatásáig eltelt időszak óriási változást mutat az információ megszerzésének lehetősége és annak gyorsasága tekintetében. Míg korábban a gyógyító orvosnak azt kellett megtudnia, hogy az őt érdeklő témában jelent-e meg publikáció, s ha igen, akkor hogyan jut hozzá, ma már az orvostudomány területén a tudomány egyéb ágaihoz hasonlóan az orvosnak az a gondja, hogyan tudja kiemelni az információ áradatából a számára legfontosabbat.

Ezen a területen még nagyobb felelősség terhel bennünket, akik adatbázisokat építünk, ugyanis ezen „építmények” által olyan releváns információt is szolgáltatathatunk az orvostudomány területén tevékenykedő gyógyító-ellátó szakembereknek, ami eddig nem volt számukra elérhető.

Diplomamunkámban az adatbázis-építést az alábbi lépések szerint végzem:

1. Az elemzés, tervezés, modellezés megtörténik, így előáll egy séma.
2. A megfelelő eszközrendszerrel leírom a sémát. Ez megadja az üres fizikai adatbázis szerkezetét.
3. Fizikai adatbázis feltöltése a szerkezetnek megfelelően.
4. Lekérdezések

4 ADATGYŰJTÉS és ELEMZÉS

4.1 Fogalmak tisztázása

Az adatbázis feltételezi az „adatszerű” kezelést, vagyis hogy az adatok mezőkbe, rovatokba rendezetten legyenek elhelyezve benne. Ehhez pedig fogalmi modellt kell alkotni. Az adatot valamilyen névvel ellátott „adatdoboz”-ként képzeljük el, melyben a név a doboz tartalma lesz. Csak hogy ehhez először magukat a neveket, az alapfogalmakat kell tisztázni. A fogalomalkotás tehát az ismeret születésének egyik és egyben első állomása.

A névadás azonban a fogalomalkotásnak csak az egyik oldala. Fel kell mérni a fogalom terjedelmét és értékészletét is. Majd az elvileg tisztázott fogalmakat az értékek ábrázolásával kell párosítani ahhoz, hogy a fogalomból adattétel születhessen. Ilyenkor felmerül a kérdés, hogy természetes vagy kódolt adatértékeket alkalmazunk-e. Ezeket mind-mind tisztáznunk kell ahhoz, hogy ne csak fogalmaink legyenek, hanem adatféléink is.[3]

Mielőtt elkezdtem az adatbázis-építést, az immunológia osztályon folyó minden folyamatot meg kellett értenem ahhoz, hogy az alapvető fogalmakat, valamint a folyamatok orvosi kifejezéseit értelmezni tudjam. Ebben csak orvosok segíthettek. Az együttműködés közben életszerűen nyilvánvalóvá vált, hogy két különböző szakterület képviselői közötti kommunikáció nem is olyan egyszerű feladat.

A szakterület képviselői az adott szakterület implicit módon beépülő fogalmaival dolgoznak, amelyeket nem biztos, hogy explicit módon meg tudnak fogalmazni. Még akár a hasonló szakterület szakemberei sem ugyanazt a nyelvet beszélik, könnyedén „elbeszélnek” egymás mellett. Az informatikusok, akik saját szakterületükön saját szakmai szlenggel dolgoznak, sem feltétlenül értik meg egymást, mint például a mérnöki és a matematikai informatikus világ sem ugyanazt a nyelvet beszéli az informatikában. A mérnökök más terminológiát, más fogalomrendszert használnak. Ezt a nehézséget úgy lehet áthidalni, hogy mi informatikusok megtanuljuk az adott szakterületet nyelvezetét, terminológiáját, specializálódunk, és így akadálymentessé válik a kommunikáció.

A sok orvosi szakkifejezés értelmezése, megértése és befogadása tulajdonképpen ugyanolyan feladatot jelent, mint az idegen nyelvek tanulása során az új szavak megtanulása. Az elsajátításukhoz időt, érthető magyarázatot és minél több konkrét példát kell adni.

Az orvos munkatársak nagyon segítőkészek voltak, összeírták a folyamatokat és belekezdtek az értelmezésükbe, viszont a magyarázat szintén tele volt tűzdelve latin szakkifejezésekkel, melyek számukra mindennapos használatúak. A dokumentum egyes részei szintén „dekódolva” voltak. Ezek többségét nem is fordítják magyarra.

Egy példán keresztül mutatnám meg, milyen nehézségek adódtak. A példában a leírt folyamatrészlet alapján egy laborvizsgálat ismereteit akartam tárolni: „Thelper2 sejtarány: CD4+/IFNg-/IL4+ 0.20 % 0.0 - 5.0”.

A fenti szövegrészt olvasva ember legyen a talpán, aki elsőre megmondja, hogy a karaktorsorban meddig tart a vizsgálat neve, eredménye, normáltartománya. Ezt olvasva, értelmezve megalkottam a vizsgálati név fogalmat. Kérdésekre, hogy meddig tart a vizsgálati név, a válasz az volt, hogy mindkét karaktorsor a vizsgálat neve „Thelper2 sejtarány” és a „CD4+/IFNg-/IL4+” szintén. Aztán rákérdeztem, melyiket használják, melyikre van szükség, vagy esetleg mindkettőre. Végül kiderült, hogy a „CD4+/IFNg-/IL4+” nemzetközi név van használatban melynek mértékegység a %, a normál tartományának kezdete 0.0, a normál tartományának vége 5.0. Így tehát a fogalom adattá vált.

Végül sikerült összeírniunk egy olyan fogalomszótárt Vincze Melindával, mely az alapvető fogalmakat tartalmazza, olyan „nyelven”, hogy azt egy olyan ember is megérthesse, akinek semmi köze sincs az orvostudományhoz. A fogalomszótár a csatolt dokumentumok között található **MYOSITIS_Fogalomszótár** néven. Kiemelném, hogy a folyamatok megértéséhez szükséges alapfogalmakat tartalmazza, viszont a tartalmára vonatkozó kifejezéseket nem, mivel akkor igen terjedelmes lett volna. Azokat implicit lehet megtudni a csatolt dokumentumokból.

Mivel tervben van az adatbázishoz kapcsolódó webes felület elkészítése, emiatt is ez a rész kiemelt jelentőséggel bír, hiszen a leendő informatikus szakember számára, aki ezt el fogja végezni, nagyban megkönnyíti majd az adatbázis fogalmainak megértését.

4.2 Interjúk készítése.

A tervezés második állomása az interjúkészítés. Az interjúvolás abból állt, hogy a leendő felhasználókat, vagyis az osztályon dolgozó munkatársakat kérdeztem arról, hogyan dolgoznak a meglévő adatbázissal, milyen a munka az osztályon, hogyan folyik a betegfelvétel, milyen adatok elérését tartanák fontosnak.

A személyes interjú annak feltárásában hasznos, hogy mit gondol a megkérdezett személy a saját ismeretigényeiről. Ha lehet, minden potenciális leendő adatbázis felhasználót ki kell kérdezni, mivel minden felhasználó szemszögéből látni kell azt, hogy milyen ismeretekre van szükségük. Ezáltal egy globális képet kaphatunk, de ehhez széleskörű egyeztetés szükséges. Ez a munka viszont nagyon időigényes is lehet.

Két személlyel végeztem az interjút, akik a klinika munkájában aktívan résztvevő munkatársak, leendő orvosok. Továbbá az elkészült papír alapú összesítéseket a főorvosnő is átnézte és jóváhagyta. Ők kapták meg a hagyatéki adatbázist, mivel az a munkatárs, aki készítette, már nem dolgozott velük. Ők viszont nem tudták használni, emiatt volt szükség az új adatbázisra és az én munkámra.

Nagyon fontos az interjú során az elsőre aprónak tűnő, lényegtelen dolgokra is rákérdezni, mivel ha valamilyen apró hibát ejtünk, átsiklunk valamilyen lényegtelennek tűnő részlet felett, amiről csak később derül ki a szerepe, ez kihatással lehet a végeredményre.

Több olyan hibába is belefutottam, mely később kihatással volt az adatbázis modellre. Az egyik a panasz tünet „keveredés” volt. Az interjúk folyamán a munkatársak hol beteg panaszokat, hol pedig tüneteket említettek. A leírásokban is keverve használták e két fogalmat, számomra következetlenül. Emiatt elkönnyveltem, hogy a tünet és a panasz fogalom számunkra egyenlő tartalmat takar. Aztán a fogalmak tisztázásakor kiderült, hogy a kettő korántsem azonos. Legalábbis számunkra csak a tünetek fontosak, a panasz nem. Viszont csak a beteg panaszok a hozzáférhetőek a zárójelentésekben. Eddig csak a panaszok voltak tárolva, emiatt migrációkor „át kellett fordítani” a panaszokat „tünet”-re, továbbá az előzetes modellezéskor változtatni kellett ezen egyedek helyén.

Aztán a következő probléma a tünetek tünetcsoportokba való besorolásakor jelentkezett. Először arra a kérdésemre, hogy erre szükség van-e az új adatbázisban, az volt a felelet, hogy „Erre nem lesz szükség”. Akkor úgy értettem, hogy egyáltalán nincs rá szükség, aztán pár hét

múlva kiderült, hogy mégis szükség van rá: „Csak be kell vinni az összes tünetet. A kritériumtünetek ebben már benne lesznek, azt nem kell külön rögzíteni. Azt mi tudjuk, mik tartoznak a kritériumtünetek közé, majd a lekérdezésnél lesz ez fontos, hogy a betegek közül kinek volt meg az összes kritériumtünete.” Megszokták ugyanis az orvosok, hogy „ránézésre” kiválogatják a kritériumtüneteket a tünetlistákból. Nincsenek hozzászokva ahhoz, hogy mindezt egy adatbázisbeli lekérdezés könnyedén megoldja helyettük.

Arra is rájöttem, hogy saját magunknak következtetéseket levonni nem szabad. Főleg az orvostudományban, ahol nem minden folyamat törvényszerű. Van, hogy a sokéves tapasztalat függvénye a beteg ápolásának következő fázisa, nem pedig egy általánosított folyamatként kell elképzelni. Úgy gondoltam, hogy a vizsgálati eredményekből és a tünetekből következik a diagnózis. Aztán kiderült, hogy néha az orvos hamarabb ismeri a diagnózist, mint ahogyan a tünetek jelentkeznének.

Amikor az interjúkkal elkészültem, a következő lépés a mezők kezdeti listájának összeállítása volt. A számított mezőket a vázlataimból kigyűjtöttem. Ez a lista tartalmazza az alapvetően szükséges adatokat, és jó kiindulópontot ad az új adatbázis összeállításához. Az interjúk alkalmával megnéztem milyen nyomtatványok vannak forgalomban az osztályon, hogyan folyik az adatok begyűjtése és megjelenítése (jelentések). Továbbá a munkatársak segítettek a központi betegnyilvántartó programhoz hozzáférni, hogy láthassam ott milyen az adatmegjelenítés.

Továbbá a klinikán dolgozó egyik munkatárstól kaptam egy részletes leírást arról, hogyan is folyik egy beteg felvétele. Ez hasznos alapként szolgált az alaptémakörök megalkotásakor, továbbá ebben segített még az általa rendelkezésemre bocsátott különböző Excel fájlok, melyeket ő használt, gyűjtött ki az adatbázisból különböző szempontok szerint.

Ezek alapján témaköröket állapítottam meg. A fő témakörök az alábbiak lettek: páciens személyi adatai, tünetek, betegségek, vizsgálatok, terápiák, projektek, családi anamnézis, rizikófaktor, teszt. Ezek mentén vizsgáltam és építettem fel az adatbázis modellt.

Azt tapasztaltam, hogy a felhasználók nem igazán tudják mit is várhatnak el egy adatbázistól, illetve mit is lehet elvárni egy számítógépes rendszertől. Ez pedig általában az információs rendszerek funkcionalitásának rohamos és követhetetlen bővülésével van összefüggésben.

4.3 A „hagyaték” elemzése.

Az adatbázis-tervezés folyamatának újabb állomása a meglévő adatbázis elemzése volt. Az osztályon dolgozó munkatársak rendelkezésemre bocsátottak egy adatbázist, mely egy Access.mdb fájl.

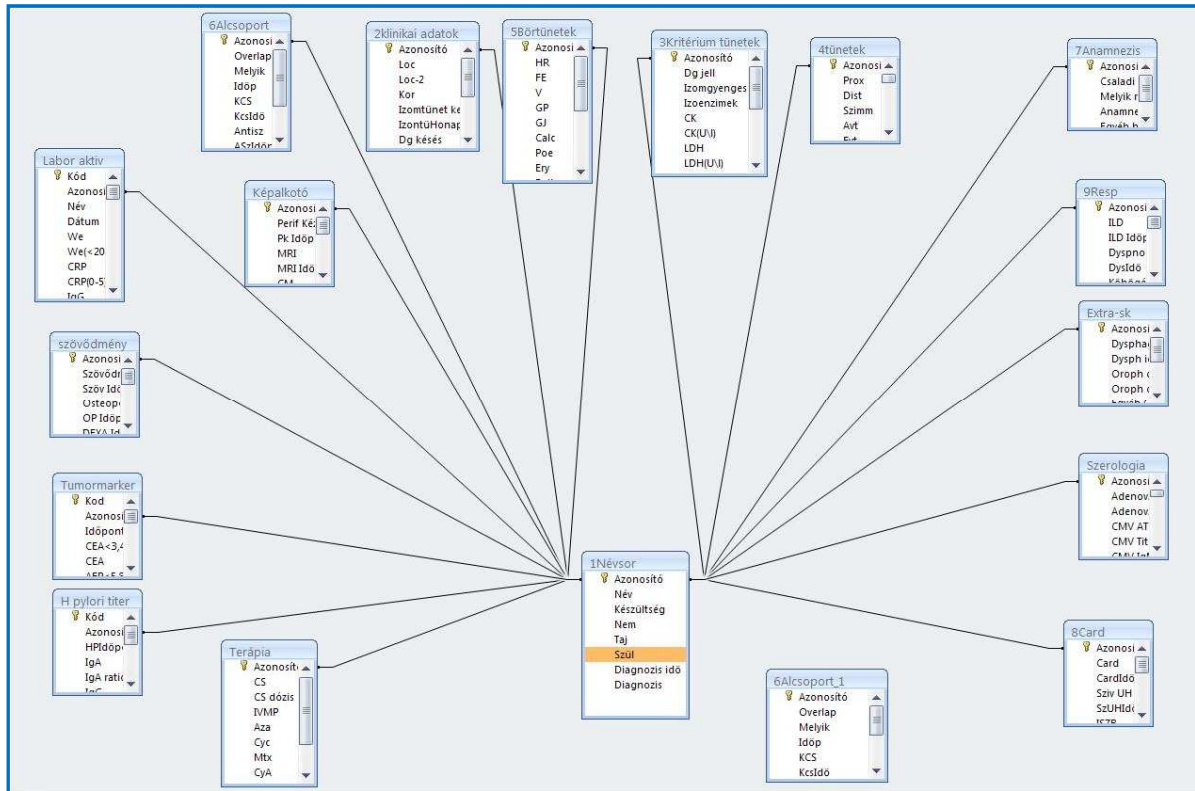
Az adatbázis 20 táblából áll. Ezek sorban: **'1Névsor', '2klinikai adatok', '3Kritérium tünetek', '4tünetek', '5Bőrtünetek', 6Alcsoport, 7Anamnezis, 8Card, 9Resp, Enzimek, Extra-sk, H pylori titer, Képkalkotó, Labor aktív, nyomtatni, Röntgendiagnosztika, Szerologia, szövődmény, Terápia, Tumormarker** (a nevek a valóságnak megfelelnek).

Az adatbázist az osztály egy régebbi dolgozója (egy PhD hallgató) állította össze, vagyis nem informatikus szakember. A felépítésében csak az játszott szerepet, hogy minden rendelkezésre álló adat szerepeljen benne. Továbbá fő szempont volt, hogy mindig az éppen aktuális probléma, kutatás adatai egy helyre kerüljenek (leggyakrabban egy táblába). Ha éppen egy kutatáshoz kellett valamilyen szempont szerint, hogy egyes vizsgálati eredmények milyen értéket vettek fel az egyes betegeknél, akkor létrehoztak egy táblát neki. Ez pedig az adatok felesleges ismétlődését vonta magával. Vagyis nagyfokú redundancia jellemzi az így elkészült adatbázist.

Másik nagy problémája a táblák közötti kapcsolat hiánya. Tulajdonképpen én emiatt nem is tekinteném adatbázisnak, csak egy adathalmaznak, amiből semmilyen új információ nem nyerhető, hiszen a táblák közötti kapcsolat nem volt megvalósítva, illetve helytelen módon volt megvalósítva, mivel minden tábla a személy azonosítón keresztül állt összeköttetésben. A hagyaték adatbázis rossz szerkezetét mutatja a 2. ábra.

Viszont azt beláttam, hogy bármilyen is ennek az adatbázisnak az állapota, típusa vagy felépítése, elemzése mindenképp sok hasznos adattal szolgálhat számomra az osztályon belül alkalmazott adatkezelési és használati módszereket illetően.

Az interjúk után kialakított fő témakörök mentén elkezdtem a hagyaték adatbázis elemzését.



2. ábra: A hagyaték adatbázis táblái

Páciens

A hagyaték adatbázisban a páciens személyi adatok eredetileg az '1Névsor' táblában tárolódtak, ami a következőképpen nézett ki.

	Azonosító	Név	Készültség	Nem	Taj	Szül	Diagnózis idő	Diagnózis
+	1	Mézga Géza	adatb	1	000-000-001	1958.02.26.	2001.05.08.	PM
+	2	Hoho Horgász	adatb	1	000-000-002	1960.06.09.	1989.09.22.	DM
+	3	Maci Laci	0	1	000-000-003	1954.01.10.	2005.04.28.	PM
+	4	Süsü Sárkány	0	0	000-000-004	1962.09.16.	2005.10.29.	PM
+	5	Frakk Kutya	0	1	000-000-005	1965.01.21.	2004.05.04.	PM

3. ábra: 1Névsor tábla

- **Azonosító** – 1-től kezdődő, az adatbázis-készítő által kiosztott egyedi azonosító.
- **Név** – A páciens neve.
- **Készültség** – adatb = egy korábbi Excel adatbázisból importált betegek.
- **Nem** – A páciens neme. Értéke 1-ha az illető személy nőnemű, 0-ha hímnemű.
- **Taj** – A páciens taj-száma. Formátuma: „XXX-XXX-XXX”.
- **Szül** – A páciens születési dátuma, „ÉÉÉÉ.HH.NN” formátumban megadva.
- **Diagnózis idő** – A myositis diagnózisának ideje napi pontossággal.

- **Diagnózis** – Maga a fődiagnózis kódja az alábbi formában: PM, DM, OM...

A következő vizsgált tábla, tartalmazza a páciens személyi adataihoz kapcsolódó adatokat, a '2**kl**inikai adatok' megnevezéssel. Felépítése:

Azonosító	Loc	Loc-2	Kor	Izomtűnet kezdete	IzontüHonap	Dg kés	Börtűnet kezd	Börtü Idő	BörtDgKésés	NDC tartama	Utolsó megj	Follow-üp
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	29			0,0			0,0	0		
3	<input type="checkbox"/>	<input type="checkbox"/>	50	2004.12.05. december		4,1			0,0	0		
4	<input checked="" type="checkbox"/>	<input type="checkbox"/>	43	2004.11.12. november		10,8			0,0	0		
5	<input type="checkbox"/>	<input type="checkbox"/>	39	2003.11.12. november		5,8			0,0	0		
6	<input type="checkbox"/>	<input type="checkbox"/>	56	2004.01.02. január		6,4			0,0	0		
7	<input checked="" type="checkbox"/>	<input type="checkbox"/>	31	1991.04.25. április		14,2			0,0	0		
8	<input type="checkbox"/>	<input checked="" type="checkbox"/>	45	1989.03.02. március		0,0			0,0	0		
9	<input type="checkbox"/>	<input type="checkbox"/>	37	1995.05.13. május		0,0	1995.05.13. május		0,0	0		
10	<input type="checkbox"/>	<input type="checkbox"/>	59	2005.10.10. október		7,2			0,0	3		
11	<input type="checkbox"/>	<input type="checkbox"/>	37	1985.07.01.		29,5	1985.07.01.		29,5	24		

4. ábra: 2**kl**inikai adatok

- **Azonosító** – 1-től kezdődő, az adatbázis-készítő által kiosztott egyedi azonosító (külső kulcs a '1Névsor' azonos nevű mezőjéhez).
- **Loc** – A páciens városban született-e? A check-box be van pipálva, ha igen a válasz.
- **Loc2** - A páciens faluban született-e? A checkbox be van pipálva, ha igen a válasz.
- **Kor** – A páciens kora, években megadva.
- **Izomtűnet kezdete** – A páciens izomtűnetei megjelenésének dátuma.
- **IzontüHonap** – A páciens izomtűnetei megjelenésének hónapja, karakteresen.
- **Dg késés** –A diagnózis felállításának időpontja és az izomtűnetek megjelenése között eltelt idő, napokban megadva.
- **Börtűnet kezd** - A páciens börtűnetei megjelenésének kezdeti dátuma, napi pontossággal.
- **Börtü Idő** – A páciens börtűnetei megjelenésének hónapja, karakteresen.
- **BörtDgKésés** – A diagnózis felállításának időpontja és a börtűnetek megjelenése között eltelt idő, napokban megadva.
- **NDC tartama** – Az NDC nevű betegség időtartama napokban megadva.
- **Utolsó megj** – A páciens utolsó megjelenésének időpontja.
- **Follow-üp** – követési idő (mióta gondozzák a beteget).

Látható volt, hogy ebben a táblában is szerepelnek a személyes adatokhoz kapcsolódó mezők: Loc-1, Loc-2, Kor. Viszont ezekről eddig nem esett szó az interjúk során. Emiatt újabb egyeztetésekre volt szükség. Rá kellett kérdeznem, hogy indokolt-e a páciensek lakcímének, és elérhetőségének tárolása. Kiderült, hogy a Cím mezőre nincs és nem is lesz szükség. Továbbá mint később megtudtam, ezen adatok rendelkezésre állnak a Klinikák által használt betegnyilvántartó rendszerben, melynek a neve Medsol. Egyébként az új adatbázist a későbbiekben egyszerű lesz ilyen módon bővíteni, egy arra alkalmas táblával.

Emiatt Loc-1, Loc-2 mezők is feleslegesek. Kiderült viszont, hogy a Kor kimaradt a felsorolásból, viszont ez a mező fontos számunkra. Azonban nem szükséges az adatbázisban tárolni, mivel ez egy számított mező, amit majd egy megfelelő nézettáblában, egy olyan kimutatásban ahol szükség van rá egy lekérdezés segítségével kiszámolunk.

Továbbá problémát jelentettek még a páciensek nevei. A hagyaték adatbázisban a Név mező szolgált a páciens nevének tárolására. Ebben előnév, férj utáni név, vezeté- és keresztnév, leánykori név egyaránt fel volt tüntetve. Az eredeti megoldás erre az lett volna, hogy mindegyik külön mezőként szerepeljen, viszont mint kiderült, a Medsol rendszerből gyakran csak pl. férj utáni nevet lehet lekérdezni, emiatt itt engedményeket kellett tennem. Megállapodtunk abban, hogy vezeté- és utónév tárolása lesz megfelelő, még ha ez nem is a legoptimálisabb tárolási mód.

Tünet

A következőkben megvizsgáltam eredeti táblacsoport: '3Kritérium tünetek', '4tünetek', '5Börtünetek'.(5. – 7. ábra) Már a nevük is utal rá, hogy ezek a TÜNET témakörhöz kapcsolódnak.

3Kritérium tünetek																	
	Az	Dg jell	Izomgyengeség	Izoenzimek	CK	CK(U\I)	LD	LDH(U\I)	GOT	GOT(U\I)	GPT	GPT(U\I)	EMG	Biopszia	Időpont	Száma	Lelet
	1	1	1	0	0	59	0	337	0	20	0	16	1	1	2001.02.21.	28812	
	2	1	1	1	0	61	0	147	x	0	x	0	1	x			0
	3	1	1	1	1	3135	1	594	0	32	0	32	1	1	2005.05.03.		0
	4	1	1	1	1	477	0	451	0	39	x	0	1	1	2005.12.07.	200500151	
	5	1	1	1	1	7320	1	21087	1	235	1	229	1	1			0

5. ábra: 3Kritérium tünetek

4tünetek	Azor	Prox	Dist	Szimm	Avt	Fvt	Nyak	Mozgásképtelenség	Izomfájdalom	Láz	Höemelkedés	Fogyás	Menny	Raynaud	Idöp	Arthralgia	AHelye
#	11	0	1	1	1	1	0	0	1	0	0	0	0	1	2001.02.01.	1	mko kézPIP,lát
#	21												0				
#	31	1	1	1	1	1	0	0	1	x	x	x	0	1	2005.09.12.	1	váll könyök jök
#	41	0	1	1	1	1	1	1	1	0	0	1	2	1	2005.10.29.	0	
#	51	0	1	1	1	1	0	0	1	0	0	0		0	0		

6. ábra: 4tünetek tábla

5Börtünetek	Azon	HR	FE	V	GP	GJ	Calc	Poe	Ery	Poik	Ptel	Fsw	Necr	Vasc	Lret	Skalp	MechK	egyéb
#	11	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	0	
#	12		1						1									
#	13	1	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	
#	14	x	x	x	x	x	0	x	x	x	x	x	x	x	x	x	0	
#	15	x	x	x	x	x	0	x	x	x	x	x	x	x	x	x	0	

7. ábra: 5Börtünetek tábla

Ezen táblák mezőit nem fejtem ki részletesen, csak nagyvonalakban térek ki egyes részeikre, mivel a táblák nagy terjedelműek (a második és harmadik tábla nem is látszik teljes egészében). Az 5. ábrán látható '3Kritérium tünetek' táblának csak egy oszlopa tünet (Izomgyengeség), a többi a vizsgálat, illetve a vizsgálatok eredményei. Itt világosan látható, hogy milyen rosszul strukturáltak az egyes táblák. A 6. ábrán látható '4tünetek' tábla. Ennek első oszlopa a páciens azonosítója (Azon), majd tünet megjelenési helyek következnek (Prox, Dist, Szimm, Avt, Fvt, Nyak). Ezt követően szintén egy tünet (Fogyás) majd ennek a mértéki jellemzője a Mennyiség oszlop látható. Tovább egy másik tünet (Raynaud), majd egy újabb jellemző (Időpont) látható. Tehát ezek alapján szembetűnő volt, hogy szükségünk lesz még minimum egy mezőre, amit első körben tünet jellemzőnek neveztem el.

A következő „maradvány” tábla a '5Börtünetek'. Itt tulajdonképpen az oszlopokban a különböző bőrtünetek rövidítései találhatóak meg. Jól látható, hogy a kialakított táblaszerkezet mennyire nem megfelelő, hiszen a tábla (sok másikkal hasonlóan) tulajdonképpen ritka mátrix felépítésű. Nagyon sok adat feleslegesen van tárolva benne. 1-es értékkel van jelölve, ha a tünet jelentkezett az adott betegnél, 0-val, ha nem és x-el, ha nem tudni jelentkezett-e. Itt jutott eszembe az a gondolat, hogy talán nincs is szükség arra, hogy melyik tünet nem jelentkezett adott páciensnél, vagy nem tudja pontosan megmondani a páciens. Egyeztettem a klinika munkatársaival, és megerősítették, hogy csak a jelentkezett tünetekre vagyunk kíváncsiak. Ez nagyban leegyszerűsíti a tünetek kezelését, illetve tárolását.

Visszatérnék pár mondat erejéig a '2klinikai adatok' táblára. Itt ugyanis volt olyan oszlop, ami a Tünetek táblánkat érintheti. Izomtűnet kezdete, Börtűnet kezd, IzontűHonap, Börtű Idő. Itt a tünet csoportokról van szó. Az első két oszlop szükséges az új adatbázisban is, az utolsó kettő viszont számított oszlopok. Ezt nem tároljuk explicit az adatbázisban. Esetleg ha később

szükség lenne rájuk (ahogy ezt már korábban is kifejtettem), akkor egy nézettáblával kezeljük a problémát.

Családi anamnézis

Azonosító	Családi anam	Melyik rokon	Anamnézis(mozgásszervi)	Egyéb betegségek	Műtétek
125 5		gyermek	alagút sy-2000. febr.	bo. pleuropneumonia-2001. dec.	agyi aneurysma ruptura miatt op.-1991. /
126 0		0	0	allergia-2002. Aug.	0
127 2		anya, unokatest	coxarthrosis-2000. szept. / alagút sy-2001. aug.	hepatitis-1970. / hyperthyreosis-2002. máj.	méhnyak poly eltáv.-1983. / tonsillectomia-1986. / cholecystectomy-1997.
128 4 8		testvér ?	jo. alkar tendovaginitis-1993. júl.	deg. pigm. ret.-1977. / struma diff. non-toxica-1998. márc. / vasculitis-2000. jan. / erythema multif.-2001. dec.	tonsillectomia-1977.
129 8		szülő	0	0	0
130 0		0	0	0	0

8. ábra: 7Anamnézis tábla

A családi anamnézis a családban előforduló „mindennemű” betegség, melyet a páciens esetleg örökölhét. Mivel a betegség kialakulásáról nincs az orvostudománynak pontos ismerete, ezért fontos ezen ismeretek tárolása.

Eredetileg a 8. ábrán látható ’7Anamnézis’ nevű táblában tárolódtak a családi anamnézishoz kapcsolódó adatok. A ’Melyik rokon’ oszlop tartalmának tanulmányozása során tűnt fel, hogy ott néha a rokoni kapcsolatok előtt a rokoni ág (anyai vagy apai) is szerepel. Így derült ki az utólagos megbeszélések során, hogy ez is fontos adat. Ezt is tárolnunk kell az új adatbázisban.

Vizsgálatok

Ez az egyik legfontosabb és legösszetettebb témakör, amely abból is látszik, hogy az adatbázisban a legtöbb tábla ehhez kapcsolódik: ’H pylori titer’, ’Képkalkotó’, ’Labor aktív’, ’Extra-sk’ stb. (9. ábra, 10. ábra). Ennek a témakörnek a megértése és kidolgozása igényelte a legtöbb körültekintést.

Az osztályra, amikor megérkezik egy beteg, különféle vizsgálatokra van szükség ahhoz, hogy pontos kórképet kaphassanak az állapotáról. Ezen vizsgálatok legnagyobb részét más munkahelyen végzik, s szöveges eredményt kapnak róla az osztályon. Vannak olyan vizsgálatok melyeket többször is el kell végeztetni, viszont van olyan, amit csak egyszer. A különböző típusú vizsgálatoknak ráadásul különböző típusú eredményei lehetnek.

Vannak vizsgálatok és vizsgálati csoportok. A páciens elküldik egy labor vagy képképző vizsgálatra (esetleg többre), akkor erről a vizsgálatról elsősorban azt tudjuk, hogy van egy vizsgálati *neve, időpontja és eredménye*. Az eredmény „jósága” vizsgálat típusonként különböző.

Vizsgáljuk meg a 'H pylori titer' táblát. (9. ábra)

Kód	Azono	HPIdőpont	IgA	IgA ratio	IgG	IgG(U/ml)	CagA IgG(U/ml)	CagA Ij
0003	3	2005.04.28.	1	18,60	1	16,80	0,50	0
0004	5	2004.10.12.	0	0,40	0	1,60	0,10	0
0005	6	2004.09.07.	0	2,70	1	18,00	1,00	0
0006	8	2004.03.19.	1	28,40	0	115,70	5,00	0
0007	10	2006.01.27.	1	55,50	1	51,70	36,00	1
0008	9	2000.10.30.	0	0,81	1	31,80	0,00	
0009	11	2003.10.13.	1	41,40	1	14,60	4,20	0

9. ábra: H pylori titer tábla

- **Kód** – A vizsgálat kódja. Egyéni 4 jegyű azonosító.
- **Azonosító** – A páciens azonosítója.
- **HPIdőpont** – A H pilori titer időpontja.
- **IgA** – Az IgA nevű laborvizsgálat eredménye, 1-es értéket vesz fel, ha pozitív, 0-át ha negatív lett az eredmény az adott páciensnél.
- **IgA ratio** - Az IgA nevű vizsgálat konkrét eredménye.
- **IgG** – Az IgG nevű laborvizsgálat eredménye, 1-es értéket vesz fel, ha pozitív, 0-át ha negatív lett az eredmény az adott páciensnél.
- **IgG(U/ml)** – Az IgG nevű vizsgálat konkrét eredménye.
-

A tábla felépítése a következő elvet követi: minden sor egy vizsgálati kóddal, illetve egy páciens azonosítóval kezdődik, a további mezők pedig felváltva egy vizsgálat nevét, e vizsgálat eredményét, majd egy másik vizsgálat nevét, vizsgálat eredményét/ét. tartalmazzák.

Ez a felépítés különösen akkor látványos, mikor 52 oszlopnyi vizsgálat – eredmény páros váltogatja egymást (pl.: '**Labor aktiv**')

A fő hiányosság abban mutatkozik meg, hogy ha nem tudunk pl. vizsgálatra keresni, mivel több táblába is le van tárolva ugyanaz a vizsgálat. Eredmények összehasonlítása szintén nehézkes vagy egyáltalán nem kivitelezhető.

	Az	Perif Kézscsa	Pk Időpont	MRI	MRI Idő	CM	Cmidöp	KézRT	KézRTGIdő	LábRTG	LábRTG Idő
+	1	1	2001.05.15.	x		x		x		x	
+	2	x		x		x		x			
+	3	x		0	1999.03.06.	1	2006.02.14.	x		x	
+	4	x		0	2005.12.06.			x		x	
+	5	x		x		x		x		x	
+	6	x		x		x		x		x	
+	7	x		x		x		x		x	
+	8	x		x		x		x		x	
+	9	x		1	2003.04.27.	0		x			

10. ábra: Képkalkotó tábla

Terápia

A 11. ábrán látható 'Terápia' tábla mutatja, hogyan tárolták a hagyaték adatbázisban az ehhez a témakörhöz kapcsolódó ismereteket.

	Azonosít	CS	CS dózis	IVMP	Aza	Cyc	Mtx	CyA	IVIG	PEX	Lycurim
+	1	<input checked="" type="checkbox"/>	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
+	2	<input checked="" type="checkbox"/>	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
+	3	<input checked="" type="checkbox"/>	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
+	4	<input checked="" type="checkbox"/>	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
+	5	<input checked="" type="checkbox"/>	1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

11. ábra: Terápia tábla

- **Azonosító** – A páciens azonosítója.
- **CS** – Az alkalmazott terápia megnevezése.
- **CS dózis** – Az alkalmazott terápia dózisa.
- **IVMP** - Az alkalmazott terápia megnevezése.
- **Aza** - Az alkalmazott terápia megnevezése.
- ...

Itt az volt fő szempont tároláskor, hogy minden betegnek láthassák az éppen alkalmazott terápiáit. Viszont rákérdeztem, hogy esetleg szükség lehet-e arra a tudásra, hogy mikor milyen terápiát alkalmaztak, és mennyi ideig tartott egy-egy terápia. A válasz az volt, hogy szükség lenne rá, emiatt az új adatbázisban ezt is tárolnom kellett. A dózisiról is egyeztetünk, és ezt nem tároljuk, mivel az alkalmazott dózis meghatározása az adott betegnél egyéni tulajdonságok figyelembevételével történik, pl.: testsúly, életkor.

Az összegyűjtött és az elemzések segítségével kiegészített adatok alapján elkezdtem a tervezést, majd a modellezést, melyet a következő két fejezetben részletezek.

5 TERVEZÉS

Az adatbázis-tervezés okai.

A legfontosabb ok, amiért adatbázis tervezéssel kell foglalkoznunk az, hogy csak így biztosíthatjuk az adatbázisban tárolt adatok következetességét, épségét és pontosságát. Ha nem jól tervezzük meg az adatbázist, bizonyos adatokhoz csak nehezen férünk majd hozzá és esélyt adunk arra, hogy a lekérdezések pontatlan eredményt adjanak. A pontatlan adatok a rossz adatbázis-tervezés legártalmasabb következményei [1]. Jó adatbázis-terv pontos adatokat eredményez, javítja az adatok tárolásának hatékonyságát, és egyszerűbbé teszi az adatbázis karbantartását. Hatékony tárolás alatt azt értjük, hogy nem engedjük meg az információk redundáns tárolását, mivel ez tárhelyet pazarol, továbbá karbantartási anomáliákat okoz [9].

Tervezés során sok mindent szem előtt kell tartanunk. Mivel az adatbázisnak egyedi igényeket kell tudnia kielégítenie, ezért mindenképpen megfelelő és hatékony táblaszerkezeteket kell létrehozniuk. Gondoskodnunk kell az adatok mező-, tábla- és kapcsolatszintű épségéről. Az adatbázisnak támogatnia kell az osztályra vonatkozó lényeges működési szabályokat, híven kell tükröznie az osztály folyamatait. Nem utolsó sorban az adatbázist fel kell készítenünk a jövőbeni bővítésre.

További lényeges szempont, hogy pontosan azonosítani kell, hogy milyen adatok tárolását kívánjuk megvalósítani, azok összetettségéről, felbonthatóságáról, más adatokhoz fűződő viszonyairól kell határoznunk, de figyelni kell arra is, hogy az adatbázis hosszú távú használatából származó működési rendellenességeket már szerkezeti szinten is elkerüljük.

Az adatbázis-tervezés előnyei.

Az adatbázis-szerkezet megtervezésébe fektetett időnek hosszú távon mutatkoznak meg az előnyei, mivel nem kell állandóan a gyorsan és rosszul megtervezett szerkezetet javíthatni.

Az adatbázis szerkezete könnyen módosítható és karbantartható lesz. A mezők és a táblák módosítása emiatt nem lesz kihatással az adatbázis többi mezőjének vagy táblájának szerkezetére, s így az adatok módosítása ez által könnyen elvégezhető.

A lekérdezések is egyszerűbbek, mivel könnyű kinyerni az adatokat.

Jelen adatbázis-tervezés célja.

Minden adatbázist valamilyen cél szem előtt tartásával hoznak létre - valamilyen feladat megoldására vagy éppen egy információs rendszer részeként. A célok meghatározásával pontosan leírhatjuk, mire is szeretnénk használni az adatbázist, így behatárolhatjuk tervezési munkánk területét [1].

Az osztály munkájának sokrétű tevékenységéhez a mindennapi klinikai munkában használt sok különálló papír és elektronikus alapú fájlok halmaza már nem megfelelő. Így tehát szükség van egy olyan elektronikus adatbázisra, melyben a betegség tünetei mellett, a laboratóriumi adatok, a speciális vizsgálatok (CT, röntgen, biopszia) és a különböző projektek során gyűjtött információk együtt tárolhatók és bármikor könnyen hozzáférhetők.

Elvárások az új adatbázistól.

- Páciens személyi adatainak nyilvántartása.
- Páciens tüneteinek nyilvántartása.
- Páciens megjelenéseinek tárolása (minden eddigi megjelenést).
- A páciens megjelenéséhez kapcsolódóan az orvos által előírt vizsgálatok, vizsgálati eredmények tárolása.
- A páciens családi anamnézisének nyilvántartása.
- A páciens rizikófaktorának tárolása.
- A páciens betegségeinek tárolása.
- A betegségek következtetéséhez szükséges tünetek és vizsgálatok kapcsolatának nyilvántartása.
- A páciensen alkalmazott terápiás kezelés adatainak követése.
- A különböző nemzetközi projektekben való részvétel nyilvántartása.
- A tünet – tünetcsoport besorolás nyilvántartása.
- A vizsgálatból és tünetekből következő diagnózisok tárolása.
- A páciens egészségi állapotáról saját maga által kitöltött teszt tárolása.
- A különböző vizsgálati típusok (Genetika, Speciális vizsgálatok, Általános vizsgálatok, Laborvizsgálatok) elkülönített tárolása.
- A különböző vizsgálatokhoz tartozó eredmények tárolása.

6 MODELLEZÉS

A rendszer közös ismérv alapján összetartozó, egymással meghatározott kapcsolatban lévő elemek jól körülhatárolt együttese. A valós világ rendszerei összetettek, ezért absztrakciós módszerrel létrehozuk az adott rendszer modelljét, és a modellt vizsgáljuk. Az absztrakció során a különös, lényegtelen és időleges sajátosságoktól elvonatkoztatunk, a lényegeseket pedig kiemeljük, összegyűjtjük.[\[10\]](#)

Az adatmodellezés három szinten történik:

- Koncepcionális (vagy fogalmi) szint - teljesen absztrakt, számítógép független.
- Logikai szint - ennél a szintnél kerül be konkrétan az adatbázis kezelő rendszer.
- Fizikai szint - a teljes fizikai megvalósítás.

A koncepcionális (magas szintű) adatmodell megalkotása elméletileg és gyakorlatilag is fontos szerepet tölt be az adatbázisok tervezésének folyamatában. A koncepcionális sémák magukba foglalják az egyes adattípusokat, azok jellemzőit, valamint a köztük fennálló kapcsolatokat és korlátozásokat. Szemléletes formában, ábrákkal mutatják be a létrehozandó adatbázis logikai felépítését.

A gyakorlatban általában a magas szintű modell (pl.: ER, EER) megalkotásával kezdődik az adatbázis építés, majd ezt alakítjuk át relációs (vagyis logikai) modell tervvé. Ennek az eljárásnak az elsődleges oka, hogy a relációs modell csak a reláció fogalmát tartalmazza, és nem foglalkozik egy sor másik, a valós világhoz sokkal közelebb álló modellel. A relációs modell nagy erőssége az adatbázis-műveletek megvalósításának hatékonyságában jelenik meg. Ez az erősség az előzetes tervezéskor viszont gyengeséggé válik, és ez az, amiért általában hasznos, ha a magas szintű tervezéssel kezdjük a munkát [\[5\]](#).

Fő céloom ebben a fejezetben egy általánosan alkalmazható, magas szintű modelltől kiinduló logikai modellalkotás folyamatának bemutatása, ami az adatbázis-tervezés lelke. Mivel az egyetemi órákon az ER modellezés alapjait sajátítottam - s mivel ez az egyik leggyakrabban használt és legrégebb koncepcionális tervezési modell – így ennek bemutatásával kezdem. Kis kitérőt teszek két problémát okozó résznél, majd táblázatos formában megadom a myositis adatbázis ER modellbeli alapelemeinek felsorolását, vagyis az egyedtípus listát és a kapcsolat típusok listáját. Ezt követi napjaink legelterjedtebb logikai modelljének, a relációs adatmodellnek a bemutatása. Amit a MYOSITIS relációs adatbázissémával zárok.

6.1 Az ER modell (FOGALMI szint)

Az ER (Entity-Relationship, Egyed Kapcsolat) modellt 1967-ben Chen írja le. Ez az első szemantikus modell, mely sématervező eszközként jött létre, növelvén a séma absztrakciós szintjét a hálós és relációs modellel szemben. A modell eszközszerével létrehozott koncepcionális sémát le lehet képezni hálós, hierarchikus vagy relációs sémává, és innen kezelni lehet ezt hálós, hierarchikus vagy relációs kezelő nyelvvel.[\[10\]](#)

Az ER modellben az adatok szerkezetét grafikusán ábrázoljuk, ún. Egyed- Kapcsolat diagramként. Három alapvető eleme lehet egy ilyen diagramnak: Egyedhalmazok, Attribútumok és Kapcsolatok.

Az Egyed- Kapcsolat diagram tulajdonképpen egy gráf, melynek csúcspontjai a fenti három elem valamelyikének felelnek meg. E különböző típusú elemeket különféle alakzatokkal ábrázoljuk.

Az egyed-kapcsolat diagram megalkotása során két ponton adódtak nehézségeim. Az egyik az idő modellezése, a másik pedig az alosztályok, vagyis a hierarchiák kialakítása modell szinten.

Az ER modell és az idő

Egyetemi tanulmányaim alatt megtanultam a modellezés alapjait, de órákon nem tűnt fel, hogy csak statikus szemléletben foglalkoztunk az adatmodellel. Olyan példákat vettünk, melyek a jelenségek pillanatnyi állapotát tükrözték. Pedig a jelenségek változnak. A statikus felfogású adatbázisban a megváltozott tartalmat aktualizáljuk, vagyis a régi tartalmat az újjal írjuk felül. Viszont esetünkben ez szóba sem jöhet. Fontos például az orvosoknak tudni azt, hogy milyen vizsgálatok voltak korábban és annak mi volt az eredménye. Az osztályon dolgozó munkatárs egyik mondata jól rávilágít erre: *„Egyrészt, mert ha már volt mondjuk biopszia és negatív, akkor nem csináltatunk újat. Viszont a HRCT vizsgálatot évente szoktuk a Jo-1 antitest pozitív betegeknél nézni, és ha majd vissza akarjuk keresni, hogy mikor terjedt rá a tüdőre a betegség, fontos, hogy meddig volt negatív a HRCT.”*Tehát fontos, hogy a korábbi állapotra vonatkozó ismeret ne vesszen el.

Vannak olyan jelenségek melyek esemény jellegűek. Az esemény jellegű egyedtípusokra (pl.: vizit), nem vonatkozhat aktualizálás, mert az esemény egyszer úgy történt meg, ahogy a dolog megesett. Ezért az ilyen típusoknak mindig kell, hogy legyen egy dátum tulajdonsága, mely az esemény bekövetkeztének az időpontjára utal, ami az idő domén szerepnevéként szolgál. Ennek és az egyed összes többi tulajdonságának a tartalma mindig egyszeres és sohasem módosulhat. Emiatt a megfelelő helyeket be kellett vezetnem ilyen tulajdonság típusokat.

Felmerült bennem, hogy jó lenne valamilyen módon a páciens állapotának változását modellezni. Ennek próbáltam utánajárni, de nem találtam kielégítő módszert a változások tükrözésére modell szinten. Itt az a probléma, hogy először is tisztázni kell azt, mikor mondjuk, hogy egy páciens állapota „jobb”. Ha jobbak a labor eredményei? Vagy ha jobb a közérzete, esetleg mindkettő? Ekkor viszont mennyivel jobb? Jósági szinteket kell meghatározni. Ez viszont nem triviális feladat, még az orvostudomány szintjén sem, így emiatt ez nyitott kérdés maradt.

Hierarchia, csoportképzés a modellben.

Gyakran előfordul, hogy egy egyedhalmaz egyedei között egyeseknek olyan speciális tulajdonságaik vannak, amelyekkel nem rendelkezik a halmaz minden egyede. Így célszerű speciális egyedhalmazokat, úgynevezett alosztályokat definiálni, ahol minden alosztálynak vannak speciális attribútumai vagy kapcsolatai. Ha egy egyedhalmazt alosztályival speciális kapcsolat köt össze, azt „az-egy”(angolul *isa*) kapcsolatnak nevezzük. Ez az öröklési kapcsolat.[4]

Modellezés közben feltűnt, hogy a különféle vizsgálatok egyes tulajdonságai hasonlóságot mutatnak, míg szinte mindegyiknek van egyedi tulajdonsága is, vagyis olyan, ami nem jellemző minden vizsgálatra. Ekkor ébredtem rá arra, hogy ez egy tipikus esete a fent említett „isa” kapcsolatnak. Ennek segítségével valósítottam meg modell szinten, a vizsgálatok hierarchiájának kialakítását.

A magas szintű ER modell megalkotása a feltárt adatok, illetve a meglévő dokumentumok alapján történt, melynek elengedhetetlen kiegészítője az a feltételrendszer mely a csatolt dokumentumok között található **MYOSITIS_ER_Feltételek** néven. Maga az ER diagram (a tulajdonságtípusok feltüntetése nélkül) a függelékben látható [\[ERdiagram\]](#). Az ER modell alapelemeinek listáját a következő táblázatok tartalmazzák.

A Myositis adatbázis egyedtípus listája

EGYED_Típus Neve	Gyenge/Erős	LEÍRÁS
BETEGSÉG	E	Az orvos által felállított diagnózis.
EGYÉBVIZSGÁLAT	E	Egyéb, máshová nem sorolható vizsgálatok
ÉRTÉK	GY	A genetikai vizsgálat eredményeként születő érték
GENETIKA	E	A genetikai vizsgálat
GÉNKÓD	GY	A genetikai vizsgálat során a páciens azonosító kódja
LABOR	E	A labor vizsgálat
LABORCSOPORT	GY	A labor vizsgálat csoportja
PÁCIENS	E	Az osztályon megjelenő egyén
RIZIKÓFAKTOR	GY	A páciens károsító anyagok
ROKON	E	A páciens hozzátartozója
SPECIÁLISVIZSGÁLAT	E	A speciális vizsgálatok
STUDY	E	Az nemzetközi projektek, amiben részt vesznek a páciensek
TERÁPIA	E	A páciensen alkalmazott terápia
TESZT	E	A páciens által kitöltött egészségügyi kérdőív
TÜNET	E	A páciens panaszainak orvos által leírt változata
TÜNETCSOPORT	E	A tünetek csoportja
VIZIT	E	A vizit, melyen a páciens mint egyed megjelenik
VIZITTÍPUS	GY	A vizit típusa
VIZSGÁLAT	E	A vizsgálati fócsoport

12. ábra: Myositis Egyed típus lista

A Myositis adatbázis kapcsolat típus listája

Mindig először azon egyed típusok láthatóak kötőjellel elválasztva, melyek között a kapcsolat típus fennáll. Aztán a következő oszlop tartalmazza a kapcsolat típus nevét, majd a kapcsolat jellemzőit: számosság és szorosság (O-opcionális, K-kötelező).

Kapcsolat Típus Egyedei	Kapcsolat Típus neve	Számosság	Szorosság
BETEGSÉG-VIZIT	DIAGNOZIS	M:N	K:O
GÉNKÓD-PÁCIENS	VAN	1:1	K:O
LABOR-LABORCSOPORT	VAN3	N:1	K:K
PÁCIENS-TÜNETCSOPORT	JELENTKEZETT	M:N	K:O
PÁCIENS-VIZIT	MEGJELENÉS	1:1	K:K
RIZIKÓFAKTOR-PÁCIENS	RIZIKÓFAKTORA	M:N	K:O
ROKON-BETEGSÉG-PÁCIENS	CSALÁDIANAMNÉZIS		
SPECIÁLISVIZSGÁLAT-ÉRTÉK	FELVEHET	M:N	K:K
STUDY-PÁCIENS	RÉSZTVESZ	M:N	K:O
TERÁPIA-VIZIT	RENDEL	M:N	K:O
TÜNET-TÜNETCSOPORT	TARTOZIK	M:N	K:K
TÜNET-VIZIT	RÖGZÍT	M:N	K:O
VIZIT-TEST	KITÖLT	1:1	O:K
VIZIT-VIZITTÍPUS	VAN2	N:1	K:K
VIZIT-VIZSGÁLAT	KÉRÉS	M:N	O:K
VIZSGÁLAT-EGYÉB	isa		
VIZSGÁLAT-GENETIKA	isa		
VIZSGÁLAT-LABOR	isa		
VIZSGÁLAT-SPECIÁLIS	isa		

13. ábra: Myositis Kapcsolat típus lista

6.2 A Relációs modell (LOGIKAI szint)

E. F. Codd 1971-ben kidolgozott egy relációs adatmodellezési eszközt. Az elméletére alapozva jött létre a relációs adatmodell, és erre épülve jöttek létre a relációs adatmodellen alapuló relációs adatbázis-kezelők.[10]

A modell fogalomrendszere.[9][10]

- **Tartomány**

A D tartomány atomi értékek egy halmaza. (D a tartomány neve). Minden tartomány bármely értékéhez hozzátartozik egy jól definiált *atomi adattípus*, és egy *formátum*. A formátum mondja meg, hogy az elemek literál formájában hogyan jelenjenek meg.

- **Attribútum**

Az attribútumok különböző szerepköreit, interpretációit jelölik ki a tartományoknak.

Az A_i attribútum tehát egy *szerepkör neve*, amelyet valamely D tartomány játszik. D-t az A_i attribútum tartományának nevezzük, és $\text{dom}(A_i)$ -vel jelöljük. Az A_i attribútum értékeit tehát ebből az egy adott tartományból veheti fel.

- **Reláció séma**

Egy reláció séma alatt az $\mathbf{R}(A_1, \dots, A_n)$ jelölést értjük, ahol R a relációséma neve, A_1, \dots, A_n pedig attribútumok. A relációs séma tulajdonképpen egy absztrakció amely, egy reláció leírására szolgál.

- **RELÁCIÓ – r(R)**

Az R reláció séma r relációja (vagy $r(R)$) nem más, mint az attribútumok tartományaiból alkotott Descartes-szorzat egy részhalmaza: $r(R) \subseteq (\text{dom}(A_1) \otimes \dots \otimes \text{dom}(A_n))$. (Ebben a definícióban a NULL értéket beleértjük az A_i attribútumok tartományaiba.)

Bármely relációs sémához tetszőleges számú reláció értelmezhető.

A relációs modellben különböző megszorításokat alkalmazunk modell illetve séma szinten egyaránt. A séma alapú megszorítások a következők: tartomány megszorítások, kulcsmegszorítások és a NULL értékekre vonatkozó megszorítások, egyedintegritási megszorítások, hivatkozási integritási megszorítások.

- **Relációs adatbázisséma**

Egy relációs adatbázis séma nem más, mint relációs sémák egy halmaza valamint az integritási megszorítások halmazának együttese. (A konkrét integritási megszorításokról a csatolt INT_MEGSZ dokumentumban olvashatunk bővebben)

- **Relációs adatbázis (példány, állapot)**

Egy relációs adatbázis példány nem más, mint relációk egy olyan halmaza, amelyben minden reláció (amelyek az adott sémán értelmezettek) teljesíti az integritási megszorításokat.

A relációs adatbázis elmélete a matematika két ágán: a halmazelméleten és az elsőrendű predikátumkalkuluson alapszik. Ez a tény garantálja, hogy a relációs adatbázisokból mindig pontos adatokat kapunk. Ezek a matematikai eszközök határozzák meg a jó tervezési módszerek alapját, és a jó adatbázis-szerkezet kialakításához szükséges építőelemeket is.

Az adatbázis-kezelő rendszereket az általuk kezelt, vezérelt, valamint megvalósított logikai adatmodellek alapján osztályozzuk. Ennek megfelelően beszélhetünk relációs, objektumorientált, hálós adatbázis-kezelőkről.

Mivel az adatbázis-kezelő rendszerek nem koncepcionális sémával működnek, emiatt a megvalósításhoz szükség van az implementációs (logikai) sémára is. Ehhez a megalkotott ER modellt vesszük kiindulási alapnak, majd az **er2rel** csatolt fájlban található leképezési szabályok segítségével létrehozuk a relációs modell sémáinak halmazát. Ezzel létrejött a MYOSITIS relációs adatbázisséma (14. ábra, 15. ábra, 16. ábra).

(Az er2rel dokumentum az Informatika Intézet honlapján található [\[9\]](#)).

A Myositis relációs adatbázisséma (1. rész)

PÁCIENS

<u>pid</u>	veznév	kernév	szüldátum	taj	nem
------------	--------	--------	-----------	-----	-----

TÜNET

<u>tid</u>	tnév	megj
------------	------	------

TÜNETCSOPORT

<u>tcsid</u>	tcsnév	tcskód
--------------	--------	--------

ROKON

<u>név</u>	nem
------------	-----

STUDY

<u>sid</u>	snév	ország	kapcstartó
------------	------	--------	------------

BETEGSÉG

<u>bno</u>	bnév	btípus
------------	------	--------

VIZSGÁLAT

<u>vid</u>	vnév	végzőmhely
------------	------	------------

LABOR

<u>lid</u>	lnév	ntartkezd	ntartvége	mértékegység
------------	------	-----------	-----------	--------------

TERÁPIA

<u>tid</u>	tnév	kiszerezés
------------	------	------------

VIZIT

<u>vid</u>	vdátum	vpaciens	végzőorvos	vtípus
------------	--------	----------	------------	--------

14. ábra: Myositis adatbázis relációséma (1. rész)

A Myositis relációs adatbázisséma (2. rész)

GÉNKÓD

<u>pazon</u>	<u>genkod</u>
--------------	---------------

RIZIKÓFAKTOR

<u>pazon</u>	<u>rfaktornev</u>	jellemzők
--------------	-------------------	-----------

CSALÁDIANAMNÉZIS

<u>pazon</u>	<u>razon</u>	<u>bazon</u>	ág	banamcsoport
--------------	--------------	--------------	----	--------------

RÉSZTVESZ

<u>pazon</u>	<u>sazon</u>	érvényes
--------------	--------------	----------

DIAGNÓZIS

<u>vizitazon</u>	<u>bnó</u>
------------------	------------

TARTOZIK

<u>tazon</u>	<u>tcsazon</u>
--------------	----------------

JELENTKEZETT

<u>pazon</u>	<u>tcsazon</u>	dátum
--------------	----------------	-------

KÉRÉS

<u>vizitazon</u>	<u>vizsazon</u>	eredmény	jóság	dátum
------------------	-----------------	----------	-------	-------

FELVEHET

<u>svizsazon</u>	<u>érték</u>
------------------	--------------

LABORCSOPORT

<u>lazon</u>	<u>lcsazon</u>
--------------	----------------

15. ábra: Myositis adatbázis relációséma (2. rész)

A Myositis relációs adatbázisséma (3. rész)

EGYÉBVIZSGÁLAT

<u>evazon</u>	evnév
---------------	-------

SPECIÁLISVIZSGÁLAT

<u>vazon</u>	vnév	vcsoport
--------------	------	----------

GENETIKA

<u>gvazon</u>	gvizsnév
---------------	----------

RENDEL

<u>vizitazon</u>	<u>terazon</u>	művelet	megjegyzés
------------------	----------------	---------	------------

RÖGZÍT

<u>vizitazon</u>	<u>tünetazon</u>	ptjellemező
------------------	------------------	-------------

KITÖLT

<u>vizitazon</u>	<u>tazon</u>	válasz	százalék
------------------	--------------	--------	----------

VIZITTÍPUS

<u>vtid</u>	vtipusnév
-------------	-----------

TESZT

<u>tid</u>	kérdés	kezdőérték	végérték	tesztípus
------------	--------	------------	----------	-----------

16. ábra: Myositis adatbázis relációséma (3. rész)

6.3 A relációsémák normalizációja.

Az emberi hibázás lehetősége mindig fennáll, vagyis semmi sem garantálja, hogy a megalkotott modellben nincsenek kisebb-nagyobb pontatlanságok. Emiatt a leképezett sémákat normalizációs módszerrel ellenőrzöm le és ahol szükség van rá, ott finomítom.

A normalizáció az a folyamat, amelynek során szétbontjuk a nem megfelelő (nem kielégíthető, rossz) relációsémákat úgy, hogy az attribútumaikat több kisebb relációsémába helyezzük át [9].

A normálforma egy olyan feltétel, melyet a relációsémák kulcsai és a bennük fennálló funkcionális függések segítségével fogalmazunk meg, s amellyel megállapítható, hogy a relációséma egy adott normálformában van-e.

Relációsémák normálformái:

- első normálforma (1NF)
- **második normálforma (2NF)**
- **harmadik normálforma (3NF)**
- Boyce-Codd normálforma (BCNF), negyedik normálforma (4NF), ötödik normálforma (5NF)

A tervezés utolsó fázisaként harmadik normálformáig végzem a normalizációt, az előállt adatbázis sémán. Nem szükséges ugyanis a lehető legmagasabb normálformáig normalizálni ahhoz, hogy helyes legyen az összeállított terv. (Ez persze nem jelenti azt, hogy egyes sémák nincsenek a 3NF-től magasabb normálformában).

1NF – az a feltétel, hogy a relációséma minden attribútuma atomi értékeket vegyen fel a különálló rekordokban. Vagyis tiltja az összetett és a többértékű attribútumokat.

Egy relációséma 2NF-ben van, ha 1NF-ben van és a relációséma minden leíró (másodlagos) attribútuma teljesen függ a relációséma elsődleges kulcsától [9]. Nyilván, ha a sémában nincs másodlagos attribútum, vagy a kulcs egyetlen attribútumból áll, akkor a séma 2NF-ben van.

Egy relációséma 3NF-ben van, ha 2NF-ben van és nincs olyan leíró (másodlagos) attribútuma, mely tranzitívan függne a relációséma elsődleges kulcsától [9].

Mivel a normálformák egymásra épülnek, ezért az első normálforma meglétével kezdem a vizsgálatot a sémákon, majd sorban haladok az egyre magasabb normálformák felé.

1NF

Az ER modell leképezése után annyit már biztosan állíthatunk a sémákról, hogy azok első normálformában (1NF) vannak, tehát ennek ellenőrzésével nem foglalkozom.

A maximum két tulajdonságtípussal rendelkező egyedtípusok mindig 5NF alakúak [3], ebből következik, hogy az ezekből leképezett két oszlopos táblák is 5NF alakúak. Tehát elmondható, hogy ötödik normálformában (5NF) vannak az alábbi relációsémák:

ROKON, GÉNKÓD, DIAGNÓZIS, TARTOZIK, FELVEHET, LABORCSOPORT, EGYÉBVIZSGÁLAT, GENETIKA, VIZITTÍPUS.

2NF

A maradék relációsémák közül, azon sémákat veszem, melyeknek elsődleges kulcsa egyszerű, ekkor ezekről elmondható, hogy második normálformában (2NF) vannak:

PÁCIENS, TÜNET, TÜNETCSOPORT, STUDY, BETEGSÉG, VIZSGÁLAT, LABOR, TERÁPIA, VIZIT, SPECIÁLIS_VIZSGÁLAT, TESZT.

Most a megmaradt sémák között (melyek biztosan 1NF-ben vannak) azt vizsgáljuk meg, van-e olyan leíró attribútum, mely az elsődleges kulcs egy részétől is függ. Vagyis van-e részleges függés. A maradék relációsémák közül a CSALÁDI_ANAMNÉZIS sémában a bazon elsődleges attribútum funkcionálisan meghatározza a banamcsop leíró attribútumot, ezért kiemeljük egy külön sémába: ANAMCSOPORT (bazon, banamcsop). (Ez a séma ötödik normálformában van). A többi sémáról belátható, hogy második normálformában vannak.

3NF

Most arra a vizsgálatra térünk át, melyben a leíró attribútumok közötti függőségi viszonyokat keressük. Az alábbi relációsémák 3NF-ben vannak, mivel csak egy leíró attribútummal rendelkeznek: RIZIKÓFAKTOR, RÉSZTVESZ, JELENTKEZETT, RÖGZÍT, CSALÁDI_ANAMNÉZIS.

A maradék sémák közül a KITÖLT sémában a válsz leíró attribútum funkcionálisan meghatározza a százalék leíró attribútumot, ezért kiemelem ebből a sémából, de nem hozok létre új sémát, mivel ez tulajdonképpen számított mező, könnyen megkapható a már tárolt adatokból.

Itt jegyezném meg, hogy a fenti eljárás során, csak azon sémákra tértem ki, melyekben az eljárás változtatást indokolt. Természetesen a többi séma elemzése is megtörtént, viszont a terjedelem miatt ezt itt nem részleteztem.

6.4 Megvalósítás (FIZIKAI szint)

Megkaptuk a relációs sémák halmazát. Ezt bele kell „ágyazni” egy konkrét adatbázis-kezelő rendszer keretei közé. Mivel én MySQL relációs adatbázis-kezelő rendszert használok, emiatt MYSQL specifikus a megvalósítás. A 17. ábrán látható a páciens tábla konkrét MYSQL-beli megvalósítása, a többi tábla a **MYOSITIS_Tablamezo** csatolt dokumentumban található meg, ehhez hasonló felépítéssel.

PÁCIENS - A tábla a páciens törzs adatokat (személyes adatokat) tartalmazza.

- DBMS-beli név: **paciens**

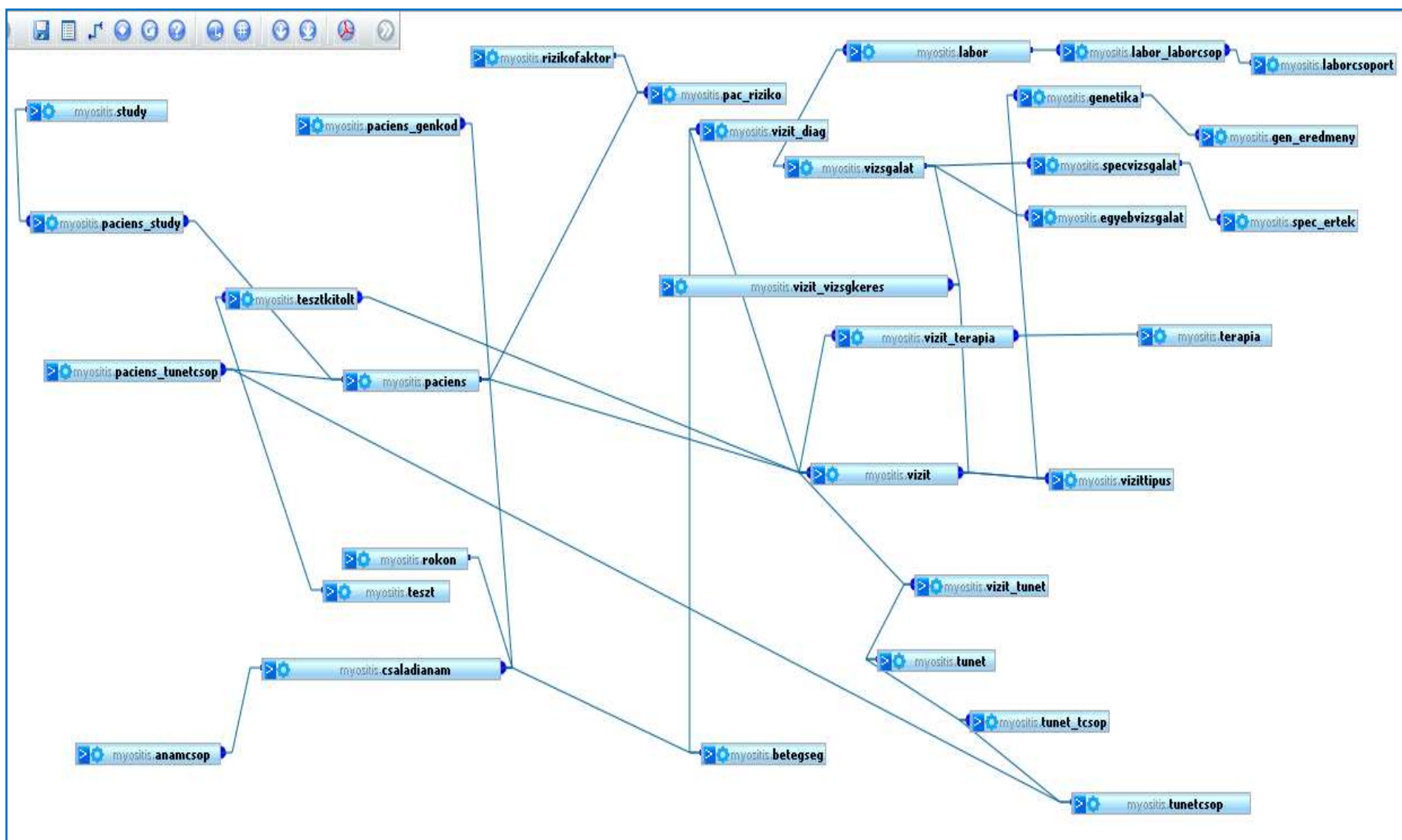
PÁCIENS				
MEZŐNÉV	KULCS	MEZŐ TÍPUS [HOSSZ]	NULL lehet-e?	LEÍRÁS
pid	PK	int (4)	NEM	A páciens azonosítója
veznev		varchar (30)	NEM	A páciens vezetéknév
kernev		varchar (50)	NEM	A páciens keresztnév
taj		varchar (11)	NEM	A páciens TAJ száma
szuldatum		date	NEM	A páciens születési dátuma
nem		enum('Férfi','Nő')	NEM	A páciens neme

17. ábra: Myositis adatbázis megvalósítás

Összességében 32 táblából áll az elkészült adatbázis. Ehhez további nézettáblák, illetve később járulékos táblák kapcsolódhatnak, melyek az adatbázishoz később kialakításra kerülő felületéhez elengedhetetlenek (pl.: session kezeléshez, naplózáshoz, felhasználók kezeléséhez stb.).

A táblákat és a táblák közötti kapcsolatokat, illetve megszorításokat létrehozó sql adatbázis script a csatolt állományok között található (**myositis.sql**).

A 18. ábrán látható a táblák phpMyAdminbeli relációs nézete, itt jól látszanak az elkészült táblák között milyen kapcsolatok vannak.



18. ábra: Myositis relációs nézete

7 ADATMIGRÁCIÓ

Addig nem láttam ennek a résznek a „nagyságát”, amíg nem körvonalazódott ki teljesen az új adatbázis szerkezete az adatok közötti összefüggésekkel és kapcsolatokkal.

Létrejött az új adatbázis séma, szeretnék lekérdezéseket végezni, csak hogy van egy nagy probléma: nincs adat, legalábbis nem ott, ahol lennie kellene. Adatmigrációra van szükség. Adatmigráció alatt értjük azt a folyamatot, amikor adatokat mozgatunk különböző alapú forrásrendszerekből a célrendszerbe. Az adatok lehetnek adatbázisban, különböző típusú fájlokban stb. Lényeg, hogy az új rendszer számára releváns információkat hordozzanak. Tulajdonképpen a migrációs folyamat megfelel az adattárházak ETL folyamatának, ami három fő részből áll:

- Extraktálás – adatok kivonatolása, kinyerése átmeneti állományokba a különböző forrásrendszerekből
- Transzformálás – a kinyert adatok tisztítása, javítása, szabványosítása, átstrukturálása.
- Loadolás – a megfelelő adatok betöltése.

Próbáltam utánajárni a szoftveres megoldásoknak, vagyis olyan szoftverek után kutattam, melyek segítségével megvalósíthatnám az ETL háromlépcsős folyamatot.

Kiderült, hogy olyan univerzális szoftver, mely a különböző típusú rendszerek közötti adatmigrációt hivatott megvalósítani, nem létezik. Nyilvánvalóvá vált, hogy bár számos adatbázis-kezelő rendszer létezik, közöttük az „átjárás” problémás feladat. Olyan egységes rendszer, amely minden egyes adatbázis-kezelő rendszer közötti adat és meta adat mozgást hivatott megvalósítani, nem létezik. Ha nem egységes rendszerben gondolkodunk, csak a számunkra szükséges két rendszert nézzük, akkor már van némi esély. Esetünkben Access adatbázis kezelő rendszerből MySQL adatbázis kezelő rendszerbe szeretnék megoldani a migrációt. Találtam erre a feladatra konvertáló programot, ami megoldaná a két adatbázis adatainak átvitelét [18]. Viszont a problémát a forrás és a célrendszer adatstruktúrája közötti nagy eltérések okozták. Emiatt az ETL mindhárom részfolyamatára szükség van.

7.1 Extraktálás

Forrásrendszer a mi esetünkben a következő három típus valamelyike:

1. papír alapú: zárójelentés, HSQ teszt, egészséget értékelő kérdőív
2. elektronikus: .xls és .doc fájlok
3. „hagyatéka adatbázis”: myositis.mdb

A papír alapú forrásokból az adatok kinyerése manuálisan történik Excel köztes állományokba. Az általam előállított előre definiált szerkezetű példaállományok alapján a helyi szakemberek töltik be ezekbe a megfelelő adatokat. Majd ezen állományokat visszajuttatták hozzám.

Az elektronikus Excel fájlok és az elektronikus dokumentumok a fentiekhez hasonló módon lettek előkészítve. Továbbá a dokumentumokból másolással lettek kiemelve az általam még fontosnak vélt adatok.

Az Access adatbázis-kezelő rendszerből exportáltam a táblákat Excel fájlokba. Ezáltal létrejöttek az eredeti táblák neveinek megfelelő Excel munkafüzetek.

Tulajdonképpen látható, hogy így egységesítve lettek a különböző típusú állományok.

7.2 Transzformálás

A következő lépés az adatok újrastrukturálása, tisztítása. Ez a szükségeknek megfelelően jelentheti mezők összevonását, formátumok változását, vagy más úton történő transzformálását is.

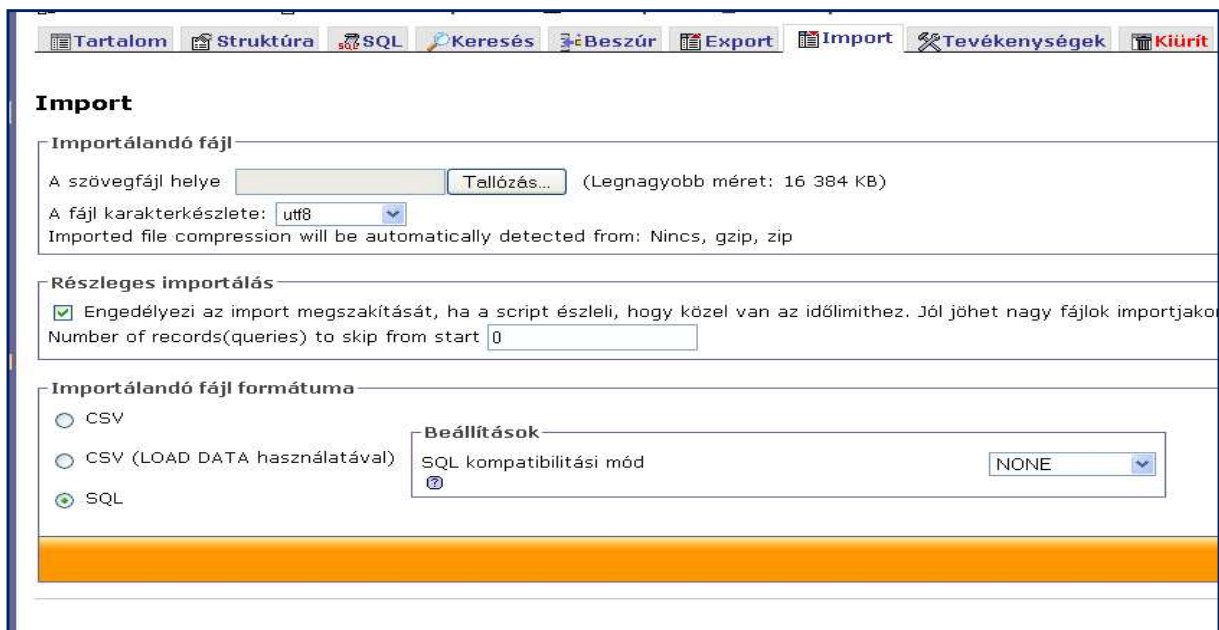
Megvannak tehát a kezdeti Excel fájlok, melyek már minden adatot tartalmaznak. Azonosítanom kellett, mely adatok szorulnak rá a módosításra, továbbá meg kellett határozni a javítás módját és mikéntjét. A módosítás érintheti az adatok típusát, tárolási hosszát vagy akár az adattartalmat is. Az adattisztítás többnyire az Excel program által biztosított funkciók felhasználásával történt. Illetve ahol lehetett ott scriptek segítségével. Mikor melyik volt célszerűbb és főleg gyorsabb. A scriptes megoldást főleg a forrásadatokból nem létrehozható tartalom előállítására vettem igénybe.

Ezen részfolyamat során is létrejöttek új, a tisztított adatokat tartalmazó fájlok.

7.3 Betöltés

A migrációs folyamat lényegi része. Az adatok betöltése a végleges adatbázis táblákba ennek részeként történik. Bár ez a részfolyamat a legfontosabb, de nehézségét tekintve már egyszerű volt az előzőekhez képest.

Maga az adat importálása a phpMyAdmin kezelőfelületének Import funkciója segítségével történt. Azokat a táblákat, amiket az eddigiek során sikerült előkészíteni, itt lehet betölteni az adatbázisba. (19. ábra)



19. ábra: Migráció / Adatimport

Probléma, hogy két fájlformátumot támogat: .CSV, .SQL. Emiatt az elkészített Excel fájlokat menteni kellett a két formátum valamelyikébe. A CSV egy gyakran használt adatsere formátum, amely minden számítógépes platformon használható. Mivel az Excel ezt a formátumot támogatja, ezért ebbe mentettem az állományokat. A fájl minden sora megfelel az Excel táblázat egy sorának, a mezők pedig pontosvesszővel vannak elválasztva egymástól.

A migrációs folyamat alatt, ha a forrás rendszer adatai közvetlenül nem, vagy csak igen nehezen tölthetők be a célrendszerbe, a köztes adatállományok használata elengedhetetlen. Ezért alaposan meg kellett fontolnom, hogy milyen fájl típust válasszak. Minden formátumnak van előnye és hátránya is. A megfelelő állomány típus meghatározza az adatkinyerés és betöltés módját. Esetünkben sok érv szólt az Excel mellett. Egyrészt a hagyaték adatbázist könnyű volt exportálni, másrészt az Excel eszközeivel könnyű volt adatokat tisztítani, majd mivel a végleges rendszer a CSV formátum „híve”, emiatt Excelből könnyű volt menteni ebbe a formátumba.

Lehet, hogy az általam választott út nehézkes, és nem is a legtökéletesebb, de figyelembe kellett venni sok tényezőt. A manuális migráció miatt a folyamat lassú és több az emberi hiba lehetőség is. Más választásunk viszont nem volt, mivel az adatok nagy része bizalmas, csak orvosi hozzáférés engedélyezett, intézményen kívül nem használhatóak. Emiatt a művelet nagyon lassú, jelenleg is folyamatban van.

8 LEKÉRDEZÉSEK, vagyis az EREDMÉNY

Adatbázist tulajdonképpen azért építünk, hogy lekérdezés igényeket tudjunk kielégíteni.

A lekérdezések által jutnak a felhasználók azokhoz az információkhoz, amire szükségük van. Tulajdonképpen, a legáltalánosabb funkciójuk lekérdezés általában különböző adatok kinyerése a táblákból. A látni kívánt adatok általában több táblában szétosztva található meg, ezeket a lekérdezések segítségével tudjuk egyetlen táblában megjeleníteni. Ezen kívül, mivel általában nem kívánjuk az összes rekordot egyszerre látni, a lekérdezésekben feltételeket adhatunk meg, amelyek „szűrik” az adatokat, így csak a szükséges rekordokat kapjuk. A lekérdezések gyakran szolgálnak rekordforrásként űrlapok és jelentések számára.

A legszélesebb körben használt relációs adatbázis kezelő rendszerek egy SQL-nek nevezett nyelv segítségével kérdezik le és módosítják az adatbázist. Az SQL a „Structured Query Language” (Strukturált Lekérdező Nyelv) rövidítése [4]. Az általam használt relációs adatbázis-kezelő rendszer is ezt használja. Emiatt a lekérdezések forráskódjait is ezen a nyelven adom meg.

A lekérdezéseket konkrét, életben felmerülő problémákat tükröznek. Minden lekérdezésnél bemutatom elsőként, hogyan lehetett az adott lekérdezést megvalósítani a régi rendszerben (ha ez egyáltalán lehetséges volt), majd az új adatbázisbeli lekérdezést adom meg az SQL nyelv és a phpMyAdmin lekérdező felülete segítségével. Mivel a lekérdezéseket teszt adatbázison végeztem, emiatt az eredménytáblák adatai csak szemléltető jellegű teszteredményeket tartalmaznak. Akkor válnak végérvényessé és ténylegesen felhasználhatóvá, ha a sikerült minden beteg adatot migrálni a rendszerbe.

Jelenleg én végzem a lekérdezéseket, vagyis közvetítő személy szükséges az eredmények és kimutatások elkészítéséhez. Emiatt a jövőben tervben van ehhez az adatbázishoz kapcsolódó webes felület létrehozása, mely az adatbevitelt, illetve a lekérdezéseket is megkönnyíti az orvosok számára.

8.1 Kérdés 1.

A Myositises betegek családjában, milyen betegségek fordulnak elő leggyakrabban?

Régi adatbázis.

Nem volt lekérdezhető, mivel a '7Anamnézis' táblának több oszlopában volt letárolva, többértékű mezőként.

Új adatbázis.

Mivel az adatbázisban csak Myositises betegeket tárolunk, ezért minden páciens részt vesz a lekérdezésben. A páciens családtagjainak betegségeit a **csaladianam** tábla tartalmazza, ebből kinyerhetjük a betegség azonosítók segítségével, azon betegségeket, melyek leggyakrabban előfordulnak. Ha ehhez hozzákapcsoljuk a **betegseg** táblát, akkor nemcsak az azonosítókat láthatjuk a kimutatásban, hanem magát a betegség neveket is.(20. ábra)

Lekérdezés 1.

```
SQL-kérés:
SELECT bazon, bnev AS BETEGSÉG_NEVE, COUNT( * ) AS GYAKORISÁG
FROM csaladianam, betegseg
WHERE csaladianam.bazon = betegseg.bid
GROUP BY bazon, bnev
LIMIT 0 , 30
```

20. ábra: Lekérdezés 1.

Eredménytábla 1.

bazon	BETEGSÉG NEVE	GYAKORISÁG
2500	Bárányhimlő	3
2507	Diabetes I.	1
3001	Polymyositis	1
3002	Dermatomyositis	1

21. ábra: Eredménytábla 1.

8.2 Kérdés 2.

A TNF alfa pozitív betegek közül hánynak van 1-es típusú diabétesze (Diabetes I.)?

Régi adatbázis.

A TNF alfa vizsgálati eredmények nem találhatóak meg a régi adatbázisban, mivel genetikai vizsgálati eredmények nem voltak benne tárolva. A betegségek viszont többértékű mezőben vannak tárolva, emiatt ez a rész nehézkes a keresés szempontjából.

Új adatbázis.

Ehhez először tisztáznunk kell a kérdést. A 'TNFalfa' egy vizsgálat neve. A 'pozitív' pedig az eredménye. 2 féle TNFalfa vizsgálat van az adatbázisban: Labor vizsgálatoknál, illetve a Genetikai vizsgálatoknál. A kérdés tisztázása után (orvosokkal) kiderült, hogy a genetikai TNFalfa-308-as vizsgálatra van szükség, azon betegeknél, akiknek pozitív lett a vizsgálat eredménye. Ez a vizsgálat egyszer van elvégezve, vagyis nem kell figyelembe venni a lekérdezésben, hogy csak a legutolsó vizsgálati eredmény szerepeljen az eredménytáblában.

Ezek alapján a lekérdezés összeállítás:

- Kiválasztjuk az I. Diabétesz diagnózis kódját

```
SELECT bno  
FROM `betegseg`  
WHERE bnev LIKE 'Diabetes I.'
```

- Kiválasztjuk a genetika táblából a TNFalfa-380 vizsgálati kódját

```
SELECT gvazon  
FROM `genetika`  
WHERE gvnev LIKE 'TNF%'
```

- A megkapott eredményeket hozzá kell kapcsolnunk a vizit_diag, a vizit és a paciens_genkod táblákhoz, hogy megkapjuk a helyes lekérdezést (22. ábra)
- A látványosabb kimutatás miatt az összesített betegszámot (összbetegszám) is meghatározzuk a páciens tábla segítségével.

Lekérdezés 2.

```
SQL-kérés:
SELECT COUNT( pg.pazon ) AS PÁCIENS_SZÁM, (
    SELECT COUNT( * )
    FROM `paciens`
) AS ÖSSZBETEGSZÁM
FROM `paciens_genkod` pg, `vizit` v, `vizit_diag` vd
WHERE pg.pazon = v.pazon
AND v.vid = vd.vizitazon
AND vd.bno
IN (
    SELECT bno
    FROM `betegseg`
    WHERE bnev LIKE 'Diabetes I.'
)
AND pg.genkod
IN (
    SELECT genkod
    FROM `gen_eredmeny`
    WHERE genvazon
    IN (
        SELECT gvazon
        FROM `genetika`
        WHERE gvnev LIKE 'TNF%'
    )
)
LIMIT 0 , 30
```

22. ábra: Lekérdezés 2.

Eredménytábla 2.

←T→	
PÁCIENS_SZÁM	ÖSSZBETEGSZÁM
2	391

23. ábra: Eredménytábla 2.

8.3 Kérdés 3.

A Jo-1 pozitív betegek közül kiknek volt meg az összes kritériumtünete?

Régi adatbázis.

A kritériumtünetek a régi adatbázisban a '3Kritérium tünetek' táblában voltak tárolva. Azonban az a probléma, hogy nem csak tüneteket, hanem vizsgálatokat is tartalmaz a tábla. Emiatt nem egyértelmű melyik oszlopban kell keresni a kritérium tüneteket.

Új adatbázis.

Két nagyobb részre lehet osztani a lekérdezést. Kik azok a páciensek, akiknek jelentkezett az össze kritérium tünete, és ezek közül kik JO-1 pozitív betegek.

- Először lekérjük azokat a tüneteket, melyek kritérium tünetek

```
SELECT `tazon` FROM `tunet_tcsop`  
WHERE `tcsazon` IN ( SELECT tcsid FROM `tunetsop` WHERE `tcskod` = 'KT')
```

- Majd, a megkapott eredményeket hozzá kell kapcsolni a vizit és a vizit_tunet táblákhoz:

Lekérdezés 3.

```
SQL-kérés:
SELECT p.veznev, p.kernev
FROM vizit_tunet vt, vizit v, pociens p
WHERE v.vid = vt.vazon
AND v.pazon = p.pid
AND vt.tazon
IN (
    SELECT tazon
    FROM tunet_tcsop
    WHERE tcsazon
    IN (
        SELECT tcsid
        FROM tunetsop
        WHERE tcskod = 'KT'
    )
)
GROUP BY v.pazon
LIMIT 0 , 30
```

24. ábra: Lekérdezés 3.

Eredménytábla 3.

			<u>veznev</u>	<u>kernev</u>
<input type="checkbox"/>			Kozs	Erzs
<input type="checkbox"/>			T	Ibolya

25. ábra: Eredménytábla 3.

8.4 Kérdés 4.

Adott páciensnek milyen vizsgálatai voltak korábban és azoknak mi volt az eredménye?

Régi adatbázis.

Ezzel lekérdezéssel az a probléma, hogy a régi adatbázisban nem volt egy helyen a vizsgálatok, több táblában van elszórva, nem egyértelmű melyikben kell keresni, továbbá a táblák csak a pácienseken keresztül vannak összekapcsolva.

Új adatbázis.

Egy tetszőleges pácienszt választva (**T. Ibolya**), megnézzük a mai napig milyen vizsgálatai voltak.

- Ehhez először a páciens táblából kinyerjük a páciens azonosítóját:

```
SELECT `pid`  
FROM `paciens`  
WHERE `veznev` LIKE 'T'  
AND `kernev` LIKE 'Ibolya'  
LIMIT 0 , 30
```

- Ezt kapcsoljuk hozzá a vizit táblához, hogy csak azon viziteket szűrjük ki, ahol a páciens részt vett.
- Majd ehhez kapcsoljuk a vizsgálat kérések és a vizsgálatok táblát, ezáltal megkapjuk a páciens vizsgálatkéréseit, továbbá az ezekhez kapcsolódó adatokat. A vizsgálat törzs táblára azért van szükség, hogy ne csak vizsgálati kódokat, hanem magukat a vizsgálati neveket is ki tudjuk listázni.

Lekérdezés 4.

```
SQL-kérés:
SELECT vnev, eredmény, josag, datum
FROM vizit_vizsgkeres vv, vizsgalat v
WHERE vv.vizitazon
IN (
    SELECT vid
    FROM `vizit`
    WHERE `pazon`
    IN (
        SELECT `pid`
        FROM `paciens`
        WHERE `veznev` LIKE 'T'
        AND `kernev` LIKE 'Ibolya'
    )
)
AND vv.`vizsgazon` = v.vid
LIMIT 0 , 30
```

26. ábra: Lekérdezés 4.

Eredménytábla 4.

←T→			
<u>vnev</u>	<u>eredmeny</u>	<u>josag</u>	<u>datum</u>
Biopszia	A pupillák tágak	Pozitív	2010-04-01
HRCT	24	Pozitív	2010-04-02
EKG	A pupscularis megnagyobbodott	Pozitív	2010-04-03
Biopszia	12	Negatív	2010-04-10

27. ábra: Eredménytábla 4.

8.5 Alkalmazott szoftverek.

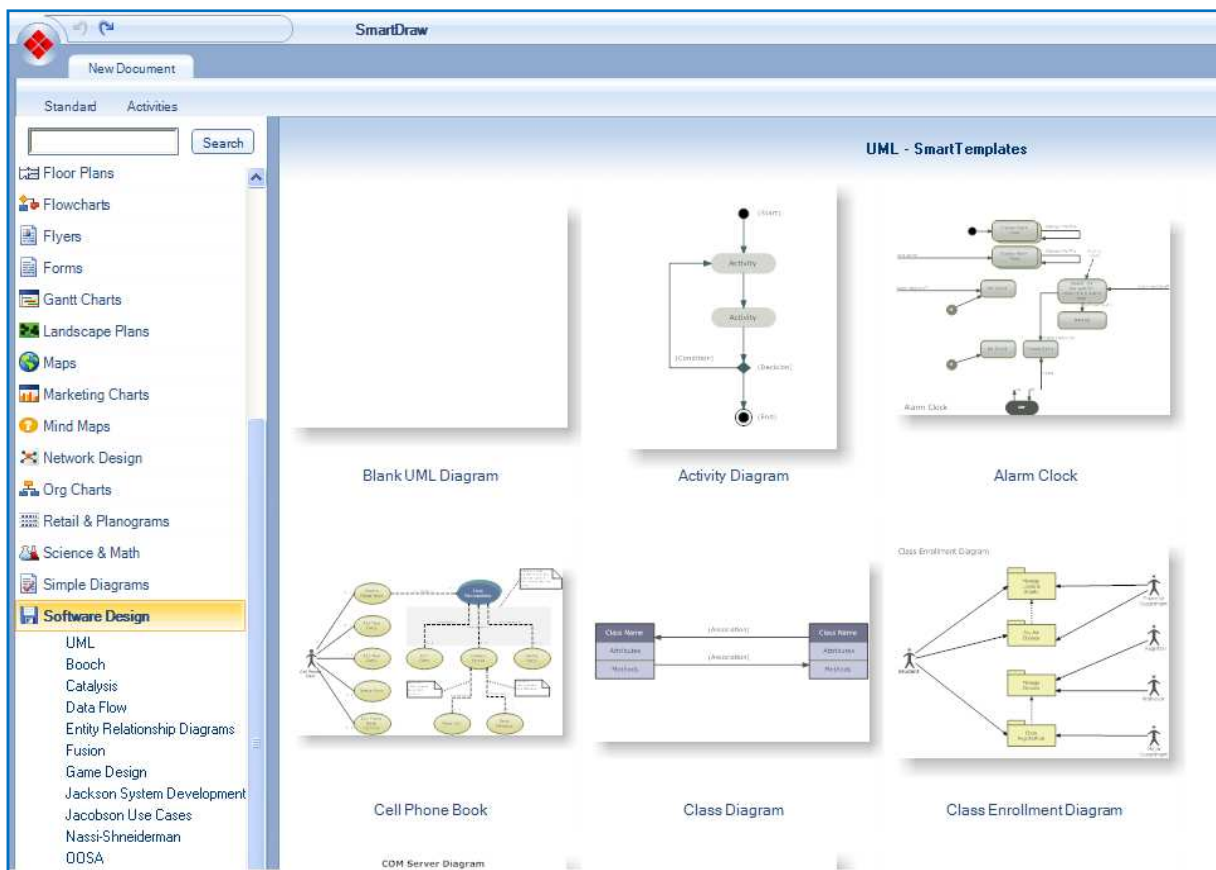
A fogalmi modell szemléltetése ER diagramokkal történik. A diagramok elkészítésében volt segítségemre a SmartDraw nevű szoftver.

SmartDraw

Maga a SmartDraw egy modellező eszköz, diagram és folyamatábra rajzoló grafikuscsomag. Használata sok előnnyel jár. Kezelése egyszerű, könnyen tanulható. Szinte percek alatt rajzolhatunk vele profi ábrákat, diagramokat, ütemterveket, műszaki rajzokat, grafikonokat.

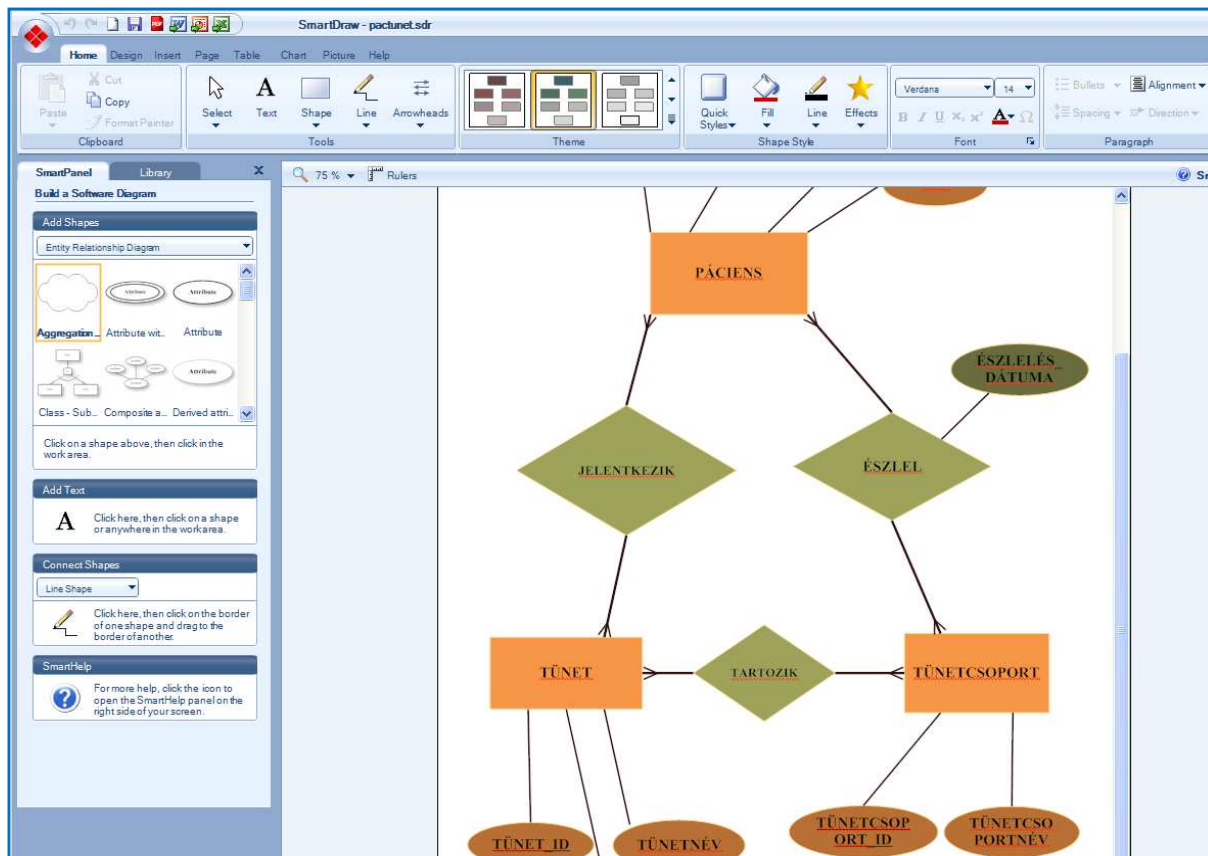
Beépített mintasablonok állnak a rendelkezésünkre (közel 1500 db.). A 28. ábrán látható néhány UML mintasablon.

Kibővített MS Office támogatás: Word-be, Excelbe és PowerPointba egy kattintással átemelhetőek az elkészített ábrák. Az adatok importja is hasonlóan egyszerű.



28. ábra: SmartDraw indítóképernyő

A SmartDraw speciális adatformátuma mellett (.sdr) bmp, jpeg és PDF adatokat képes fogadni, illetve elmenteni. A szoftvernek letölthető egy ingyenes változata, kipróbálásra. [16] Jelenleg a SmartDraw 2010-es verzió az aktuális.



29. ábra: SmartDraw ER diagram

Most bemutatom, hogyan is készítettem ezzel a szoftverrel egy ER diagramot. (29. ábra) Első lépésként ki kell választani a 28. ábrán a bal oldali menüsorban látható beépített 'Software Design' library-ból az 'Entity Relationship Diagrams' sablont. Ekkor létrejön nekünk egy üres munkalap, továbbá a szoftver a kiválasztott sablonnak megfelelő ábrákat és objektumokat a bal oldali menüsorban helyezi el. Esetünkben: Entitás, Attribútum, Kapcsolat stb. Ezek egyszerűek, nevük lényegre törő. Paneles, behúzogató módszerrel a munkaterületen ott helyezük el őket, ahol nekünk jól esik. Színezés, formázás ezek után bármelyik elemnél megengedett egyénileg, illetve csoportosan is, ami nagy segítség volt, főleg mivel az ER diagram sok apró elemből tevődik össze, így ha valamelyik részt egy nem várt pillanatban bővítenem kellett, ezáltal egyszerűen megtehettem.

XAMPP

A XAMPP egy ingyenes nyílt forráskódú web szerver csomag. Legfőbb összetevői az Apache HTTP Server, a MySQL adatbázis továbbá egy interpreter a PHP, illetve Perl programnyelveken írt scriptekhez. Ezek kezeléséhez kapunk még néhány beépített modult is, melyek közül leglényegesebb az OpenSSL és a phpMyAdmin. [12]

A csomag nagy előnye, hogy minden lényeges és főleg ingyenes kiszolgálót egyesít. Tehát egy csomagban találhatunk web-, adatbázis-, FTP- és levelező-kiszolgálót. Telepítése egyszerű, szinte semmilyen konfigurációs beállításra nincs szükség.

Pár mondatban bemutatom a XAMPP számomra lényeges komponenseit, majd kissé bővebben a phpMyAdmint, amivel a relációs adatbázis létrehozását végeztem.

Apache HTTP Server [13] - egy nyílt forráskódú webkiszolgáló alkalmazás, mely kulcsfontosságú szerepet játszott a World Wide Web elterjedésében. Olyan web szerver program, amely megfelel a gyorsan változó Internet követelményeinek, biztonságos, üzleti, vállalati felhasználásra is megfelelő és szabadon használható. Többek között a következő operációs rendszerekhez készítették el az Apache-ot: Unix, Linux, Microsoft Windows. Sok szabványt támogat, melyeknek nagy része fordított modulok formájában áll rendelkezésre a mag kiegészítéseként. Statikus és dinamikus weboldalak közzétételére egyaránt használják. Sok webalkalmazás az Apache által nyújtott környezethez és szolgáltatásokhoz van tervezve.

MySQL [14] - A MySQL egy több felhasználós, többszálú, SQL-alapú relációs adatbázis-kezelő szerver. A MySQL az egyik legelterjedtebb adatbázis-kezelő. Ennek egyik oka, hogy a teljesen nyílt forráskódú LAMP (Linux – Apache – MySQL - PHP) összeállítás részeként költséghatékony és egyszerűen beállítható megoldást ad dinamikus webhelyek szolgáltatására. Egyedi illesztő felületekkel az adatbázis-kezelő elérhető C, C++, C#, Delphi, Java, PHP és még sok más programozási nyelvvel. Egy *MyODBC* nevű ODBC interfész további, ODBC-t kezelő nyelvek számára is hozzáférhetővé teszi az adatbázis-kezelőt.

A MySQL különböző platformokon futtatható: Linux, Windows régebbi és frissebb verzióin. A MySQL adatbázisok adminisztrációjára a mellékelt parancssori eszközöket (*mysql* és *mysqladmin*) használhatjuk. A MySQL honlapjáról grafikus felületű adminisztráló eszközök is letölthetők.

phpMyAdmin[15]

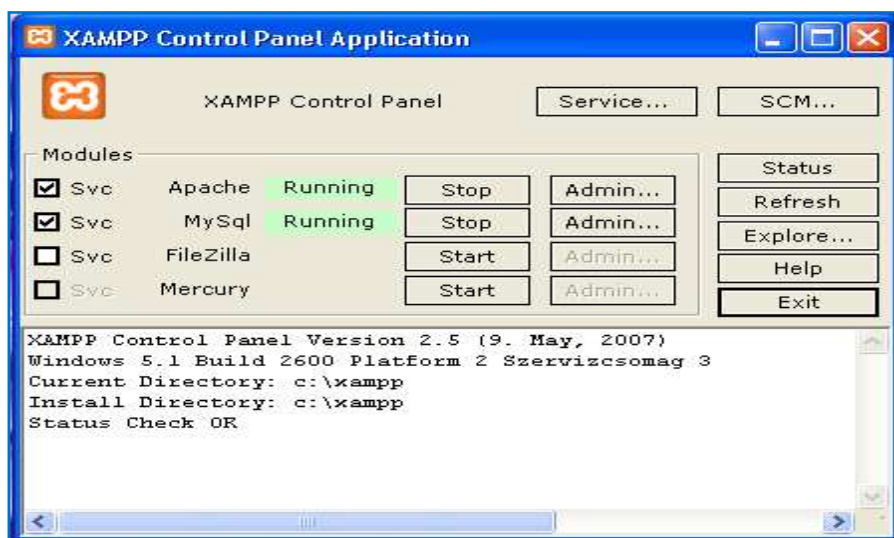
A phpMyAdmin egy nyílt forrású eszköz, amit PHP programnyelven írtak a MySQL adatbázis kezelő rendszer menedzselésére. Egy olyan adminisztrációs eszköz, mely képes készíteni és kezelni adatbázisokat, táblákat, mezőket, SQL parancsokat futtatni.

Mindegy, hogy a segítségével az egész MySQL szerveret kezelhetjük (szuper-felhasználót igényel), vagy csak egyetlen adatbázist. Az utóbbi megvalósításához be kell állítani a MySQL felhasználót, hogy csak a kívánt adatbázist tudja írni/olvasni. Jelenleg 54 nyelven érhető el.

A XAMPP telepítése, használata.

A **letöltés** és **installálás** lépései megtalálhatóak a csatolt **XAMPP_INSTALL** dokumentumban, ezért itt nem részletezem.

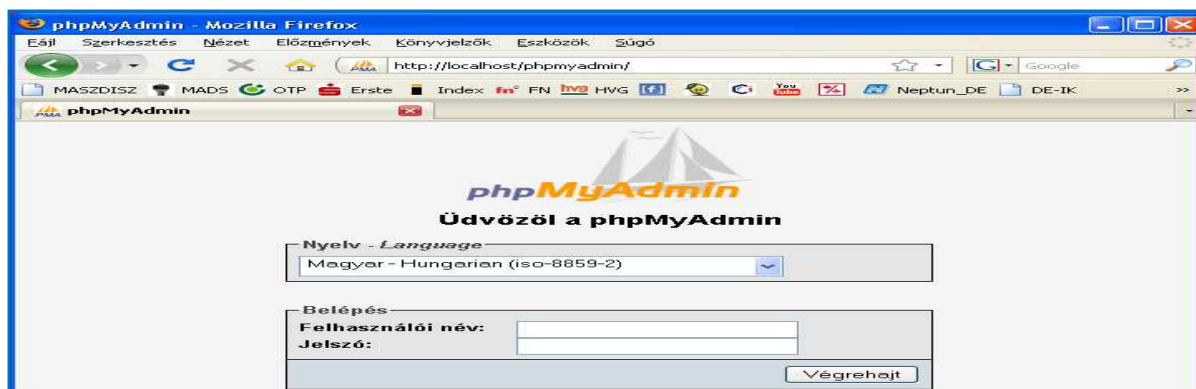
Telepítés után a XAMPP ikonjára kattintva elindul a vezérlőpult (XAMPP Control Panel), amit a 30. ábrán láthatunk. Innen irányíthatjuk frissen feltelepített kiszolgálóinkat és láthatjuk az állapotukat is. Mindegyik szolgáltatást az 'Admin...' gombon át vezérelhetünk.



30. ábra: XAMPP Control Panel

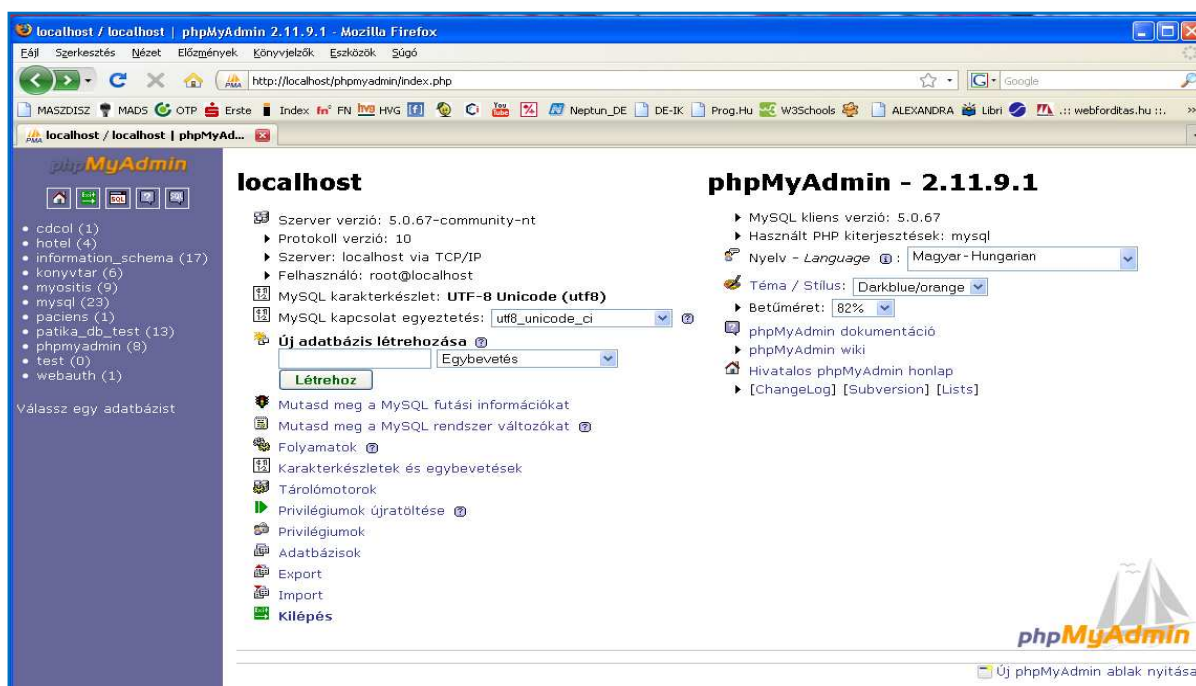
Ha tehát a XAMPP vezérlőpultján a MySQL melletti Admin gombra kattintunk, akkor megnyílik a böngésző a PHPMyAdmin kezelőjének magyar nyelvű felületével (31. ábra). A felhasználói név-jelszó párost a már említett XAMPP_INSTALL dokumentumban leírtak szerint tudjuk beállítani. Ezzel egy root jogosultsággal rendelkező felhasználót készítünk

adatbázisunkhoz, aki bármihez hozzáférhet és bármit módosíthat. A 'Végrehajt' nyomógomb segítségével hagyhatjuk jóvá belépési szándékunkat.



31. ábra: phpMyAdmin

Bejelentkezés után (31. ábra) láthatjuk, hogy két fő részre tagolódik a munkaterület. A középső nagyobb területen a MySQL szerver és a phpMyAdmin beállításokat tehetjük meg, továbbá itt tudunk létrehozni új relációs adatbázist. Középen egy bemeneti mezőbe beírhatjuk a készítendő adatbázis nevét, amit a 'Létrehoz' nyomógombbal hagyhatunk jóvá. Ezzel létrejön egy új adatbázis, mely táblákat még nem tartalmaz, de az már megjelenik a bal oldali munkaterületen, ahol is a létrehozott adatbázisok nevei vannak felsorolva. A név után zárójelben látható, hogy hány táblából áll az adott adatbázis.



32. ábra: phpMyAdmin adatbázis beállítások

Válasszuk ki a myositis nevű adatbázist, mely teszt jelleggel lett létrehozva. A nevére kattintva a 33. ábrán lévő ablak válik láthatóvá. Baloldalon az eddigi adatbázis felsorolás helyett, most a myositis adatbázis tábláit láthatjuk. Középen a különböző fülek segítségével, az adott funkciókhoz tartozó nézeteket tudjuk váltani. Most csak a számomra lényegeseket mutatom be.

SQL – ezen a panelen végeztem a lekérdezéseket, amit a Lekérdezések fejezetben részletezek.

Export – funkció segítségével exportáltam az új myositis adatbázis scriptet.

Import – funkció segítségével töltöttem be az adatbázisba azokat a táblákat, melyeket a migráció folyamán sikerült előkészíteni.

Tervező – a relációs adatbázisséma nézete. (18. ábra)

Tábla	Parancs	Sor	Típus	Egybevetés	Méret	Felülírás
<input type="checkbox"/> betegség		8	MyISAM	latin1_swedish_ci	2.2 KB	-
<input type="checkbox"/> labor		4	MyISAM	utf8_unicode_ci	2.2 KB	-
<input type="checkbox"/> paciens_diag		3	MyISAM	latin1_swedish_ci	2.0 KB	-
<input type="checkbox"/> paciens_labor		5	MyISAM	latin1_swedish_ci	2.1 KB	-
<input type="checkbox"/> paciens_tunet		4	MyISAM	utf8_unicode_ci	2.1 KB	-
<input type="checkbox"/> paciens_tunetcsop		1	MyISAM	utf8_unicode_ci	2.0 KB	-
<input type="checkbox"/> személy		185	MyISAM	utf8_hungarian_ci	12.0 KB	-
<input type="checkbox"/> tunet		2	MyISAM	latin1_swedish_ci	2.1 KB	-
<input type="checkbox"/> tunetcsop		4	MyISAM	utf8_unicode_ci	2.1 KB	-
9 tábla	Összesen	216	MyISAM	latin1_swedish_ci	28.8 KB	0 Bajt

↑ Összeset kijelöli / Összeset törli A kijelöltekkel végzendő művelet

Nyomatási nézet Adatkönyvtár

Új tábla létrehozása a(z) myositis adatbázisban

Név: Mezők száma:

Végrehajt

Új phpMyAdmin ablak nyitása

33. ábra: Myositis adatbázis phpMyAdminben

9 IRODALOMJEGYZÉK

9.1 Könyvek

1. Michael J. Hernandez: „Adatbázis tervezés”
2. Halassy Béla: „Adatmodellezés”
3. Halassy Béla: „Az adatbázis tervezés alapjai és titkai” (1994)
4. Jeffrey D. Ullman - Jennifer Widom: „Adatbázis-Rendszerek” (Alapvetés, Második, átdolgozott kiadás)
5. Timár & Társai: „Építsünk könnyen és lassan adatmodellt!”
6. Elmasri Navathe: „Fundamentals Of Database Systems”

9.2 Online irodalom

7. Database Systems - <http://infolab.stanford.edu/~ullman/dscb.html#projects>
8. Data Silence Journal - http://www.jstage.jst.go.jp/article/dsj/1/0/1/_pdf
9. Adatbázis rendszerek előadás (DE-İK) - <https://it.inf.unideb.hu/honlap/adatbazis>
10. Adatbázis rendszerek – Jegyzet, 1997 – Dr. Juhász István
11. WIKIPEDIA - <http://hu.wikipedia.org/wiki>
12. XAMPP - <http://www.apachefriends.org/en/xampp.html>
13. APACHE - <http://www.apache.org>
14. MySQL - <http://www.mysql.com>
15. phpMyAdmin - <http://hu.wikipedia.org/wiki/PhpMyAdmin>

9.3 Szoftverek

16. SmartDraw: <http://www.smartdraw.com/downloads/t/indexb.htm>
17. XAMPP: <http://www.apachefriends.org/en/xampp.html>
18. Access to MySQL: <http://www.bullzip.com/products/a2m/info.php>

10 ÖSSZEGZÉS

Diplomamunkámban egy orvosi adatbázis létrehozásának lépéseit mutatom be. Az adatbázis egy speciális és ritka autoimmun betegséghez kapcsolódó adatok tárolására jött létre.

Az alapfogalmak tisztázása és az interjúk készítése után a hagyaték adatbázist elemzem. Az elemzés rámutat jó néhány olyan lényeges tényezőre, melyek a megelőző folyamatok feltárása során elkerülte mind az én, mind az orvos munkatársak figyelmét.

A tervezés a modellezéssel folytatódik. A magas szintű ER modell megalkotása sok körültekintést igényelt, melyet az általam rögzített feltételrendszer tesz teljessé. Az ER diagram megalkotását a SmartDraw szoftverrel végeztem. Ezt a modellt a megfelelő szabályrendszer segítségével leképeztem relációs modellé. Mivel az emberi hibátényező mindenhol jelen van, ezért a leképezett sémát normalizációs eljárással finomítani kellett.

Létrejött tehát egy (legalább) 3NF-ben lévő relációs séma, ennek feltöltése tényleges adatokkal azonban sok nehézségbe ütközött, amit adatmigráció fejezetben részletezek. Relációs adatbázis-kezelő szoftvernek a MySQL-t választottam, a phpMyAdmin nyílt forrású eszköz pedig a MySQL adatbázis kezelő rendszer menedzselésében volt segítségemre. Ebben végeztem az orvosoknak szükséges lekérdezéseket, melyek mostani konkrét felmerülő problémáikra, kérdéseikre adnak választ. A lekérdezéseket az utolsó fejezet tartalmazza.

Munkám alatt megtapasztalhattam, mennyi nehézséget jelent a kommunikáció két különböző szakterület szakemberei között. Az adatbázis tervezése és építése alatt olyan életszerű problémákba ütköztem, melyeket az egyetemi órákon nem lehetett megtanulni, illetve felkészülni rájuk. Úgy gondolom, hogy sikerült nemcsak nekem, hanem az orvosoknak is egy informatikai szemléletmódot elsajátítaniuk az együtt eltöltött idő alatt.

A munka nagy terjedelmű lett, emiatt a specifikus részek a csatolt dokumentumokban találhatóak meg.

Választ adtam sok kérdésre, de mindegyik problémát nem tudtam megoldani, mivel minden egyes megoldott probléma a feltevések tucatjait vonta maga után. Remélem az adatbázis által nyert információk az orvosoknak segítenek megérteni és felderíteni azokat az összefüggéseket, melyeknek most még nincsenek az orvostudomány birtokában ezen ritka betegséggel kapcsolatban.

10.1 Köszönetnyilvánítás

Ezúton szeretnék köszönetet mondani elsősorban témavezetőmnek, **Dr. Végh János** tanár úrnak, aki szakmai tanácsaival, látásmódjával és élet tapasztalatainak megosztásával is gazdagította ismereteimet az együtt töltött idő alatt. Neki köszönhetem többek között azt a felismerést, hogy a „Meg kell húzni a határt!” az élet sok területén helytálló mondat.

Továbbá köszönetet mondok önzetlen és fáradhatatlan munkájáért **Vincze Melindának**, aki az orvosi részek feltárásában nagy segítségemre volt. Segített minden területen, ahol csak módjában állt. Továbbá hálás vagyok **Molnár Péternek**, aki szintén lelkes és nagyon segítőkész munkatárs volt az orvosi fogalmak tisztázásában és megértésében.

Külön köszönetemet fejezem ki **Dr. Dankó Katalin** professzor asszonynak mivel az ő segítsége és jóváhagyása nélkül nem jöhettek volna létre a kiegészítő dokumentumok, illetve a diplomamunka sem.

Köszönöm **Vágner Anikónak** az adatbázis-építéshez nyújtott szakmai megfigyeléseit, észrevételeit.

Külön köszönettel és tisztelettel adózok **Juhász István, Kósa Márk és Espák Miklós** tanároknak, akik az egyetemi éveim alatt munkásságukkal és szaktudásukkal értelmileg gazdagítottak és emberileg támogattak sok nehéz pillanatban.

Köszönöm végül, de nem utolsósorban **szüleimnek** a sok támogatást és a szeretet. Ők azok, akik még akkor is hittek bennem, amikor én már feladtam volna. Nélkülük, ma nem lennék karnyújtásnyira az oklevéltől.

„Két dolog van a világon, amely nem fogy el soha, ha másokkal is megosztják. Az egyik a szeretet, a másik pedig az ismeret. Ez a két dolog szorosan összefügg. Az ember társas lény. Azért kapott partnereket, hogy megtalálja azokat, akikre támaszkodhat és megkeresse a támogatandókat. Egész életünket a másokkal való kommunikáció, vagyis az ismeretek cseréje határozza meg. A kölcsönös informálás saját létünk nélkülözhetetlen feltétele, és ez egyben a szeretet által diktált kötelezettség is, mivel mások életéért is felelősek vagyunk.”^[3]

11 FÜGGELÉK

11.1 Az autoimmun betegségek és a Myositis.

Autoimmunitásnak nevezzük azt a jelenséget, amikor az immunrendszer a szervezet saját alkotóelemeit „idegen testnek” érzékeli, és elpusztításukra törekszik. Az elpusztítandó alkotóelemek (antigének) általában a létfontosságú fehérjék vagy poliszaharidok, melyek a sejtek felszínén vagy annak belsejében találhatóak. Az immunreakció során gyulladás alakul ki az érintett szövetben, és idővel a fehérjét szintetizáló sejtek, illetve a sejtek által alkotott szövet is károsodhat, végső esetben el is pusztulhat.

Idiopáthiás inflammatorikus myopáthiákban a keringésben jelen lévő fehérvérsejtek (T-sejtek) az izomzat különböző fehérjéit támadják meg. Ezen myopathiák közé tartozó polymyositis (PM), dermatomyositis (DM) és inclusion body myositis (IBM) szisztémás autoimmun betegségek, melyek közös jellegzetessége a végtagizmok (felkar, comb) krónikus gyulladása, amely fokozatosan súlyosbodó izomgyengeséghez vezet. DM esetén vöröses-lilás színű bőrelváltozások jelenléte is jellemző. A fent említetteken kívül ismerünk még tumorról társuló (CAM), gyermekkorban jelentkező (JDM), illetve egyéb autoimmun betegséghez társuló, úgynevezett „overlap myositist” (OM).

A myositisben szenvedő betegek számára nehézséget jelent a mindennapi feladatok ellátása, mint például a lépcsőn járás, fésülködés, borotválkozás, továbbá rágási, nyelési, légzési nehezítettségek jelentkezhetnek. Súlyos esetben egy időre akár mozgásképtelenné is válhat a beteg.

Az izom érintettségén és a bőrtüneteken túl a betegség előrehaladtával gyomor- bélrendszeri és légzőszervi tünetek is jelentkezhetnek, sőt súlyos esetben a betegség a szívizmot is megtámadhatja. Fájdalmas nyelés, gyomorégés, székrekedés jelezheti, hogy a gyulladás továbbterjedt az izmokról.

A betegség pontos kiváltó oka egyelőre nem tisztázott, az eddigi kutatások alapján valószínűleg környezeti hatásoknak (pl.: napfény, dohányzás, vegyi anyagok) és az öröklött genetikai tulajdonságoknak egyaránt fontos szerepe van a kialakulásában. A genetikai hatásokat igazolja, hogy a myositisek bizonyos népcsoportokban (pl.: kaukázusiakban)

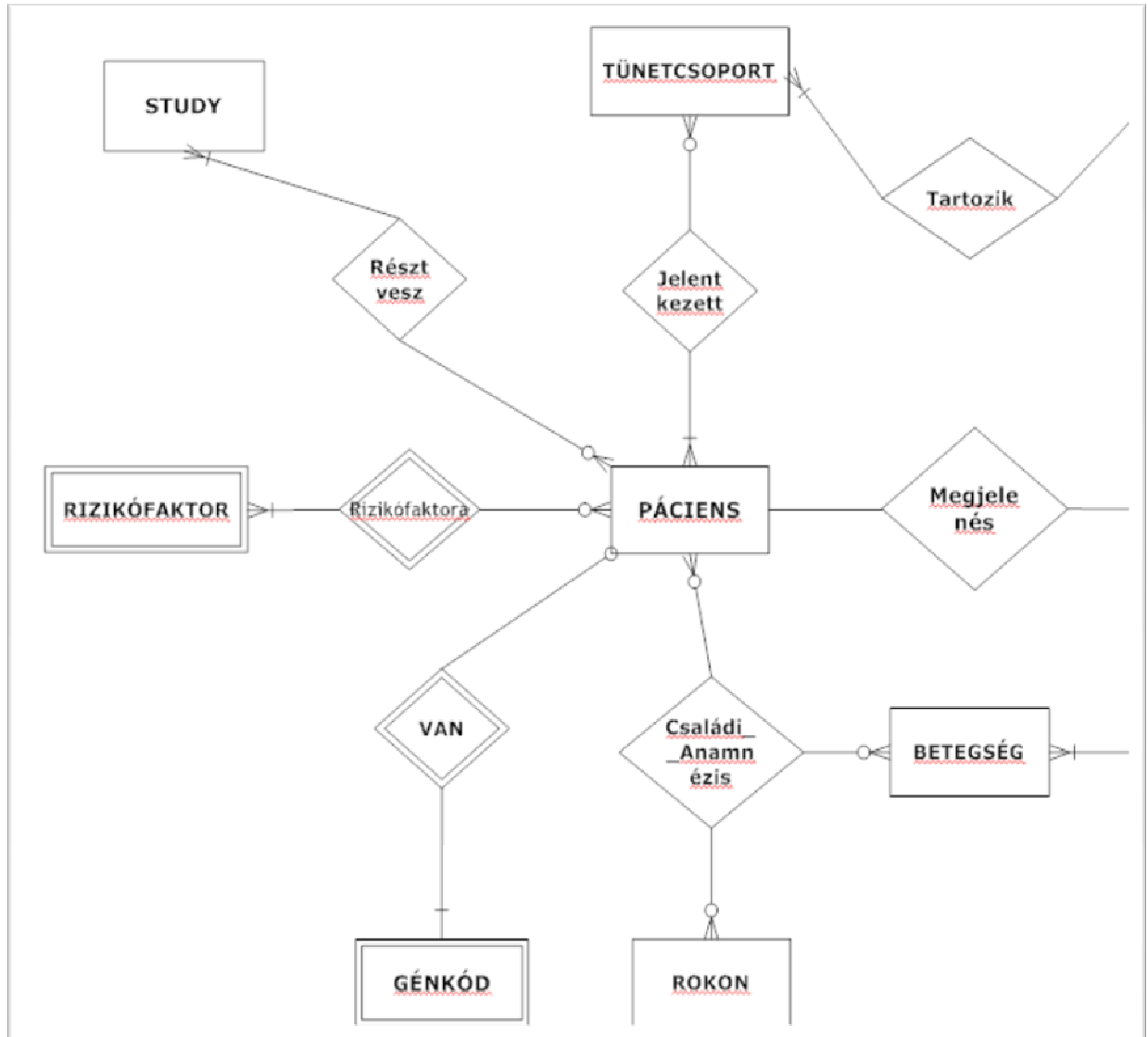
gyakrabban fordulnak elő, illetve, hogy egyes családokban „halmozódnak”, öröklődnek az egymást követő generációkban.

Ahhoz, hogy a diagnózis felállítható legyen, a fenti tünetek felismerése mellett néhány vizsgálat elvégzése is szükséges. A vérminták vizsgálatával kimutathatóak a myositisre jellemző autoantitestek, valamint olyan enzimek, amelyeknek szintje jelzi az izomkárosodás mértékét és a gyulladás súlyosságát. Ezek mellett az esetek többségében elektromyográfiai vizsgálatra (EMG) és izombiopsziára is sor kerül. A biopszia a diagnózis igazolására, az izomkárosodás mértékének kimutatására szolgál, valamint a különböző T-sejtek megfestésével a myositis alcsoportjai is elkülöníthetők.

A myositis okozta tünetek kezelés nélkül nem javulnak, sőt nagy valószínűséggel romlani fognak, ezért lényeges a mielőbbi diagnózis, és az időben elkezdett terápia. Bár a kórkép az orvosok jelenlegi tudása szerint nem gyógyítható, gyógyszerekkel és egyéb terápiás lehetőségekkel jól karbantartható. Átmenetileg vagy végleg bizonyos szintű életviteli változtatásokra is szükség lehet, hiszen a betegség fellobbanásai idején a fizikai terhelhetőség lényegesen csökken, a betegnek pihenésre, kíméletre van szüksége. A terápia mindig személyre szabott. Mind az alkalmazandó gyógyszereket, mind azok pontos dózisát, mind pedig a kiegészítő terápiás lehetőségeket egyénre szabva állapítják meg az orvosok. Ugyanakkor mivel a betegség lefolyása hullámzó, a kezelés menete idővel egy-egy beteg esetében is változhat. A gyógyszerek között mindig megtalálhatóak a kortikoszteroidok, valamint az ún. immunszuppresszív készítmények. Ezek nagyon hatásos, de ugyanakkor számos mellékhatás lehetőségét magukban rejtő gyógyszerek. Éppen ezért mind az adagolásukat, mind az egyéb orvosi utasításokat pontosan be kell tartani.

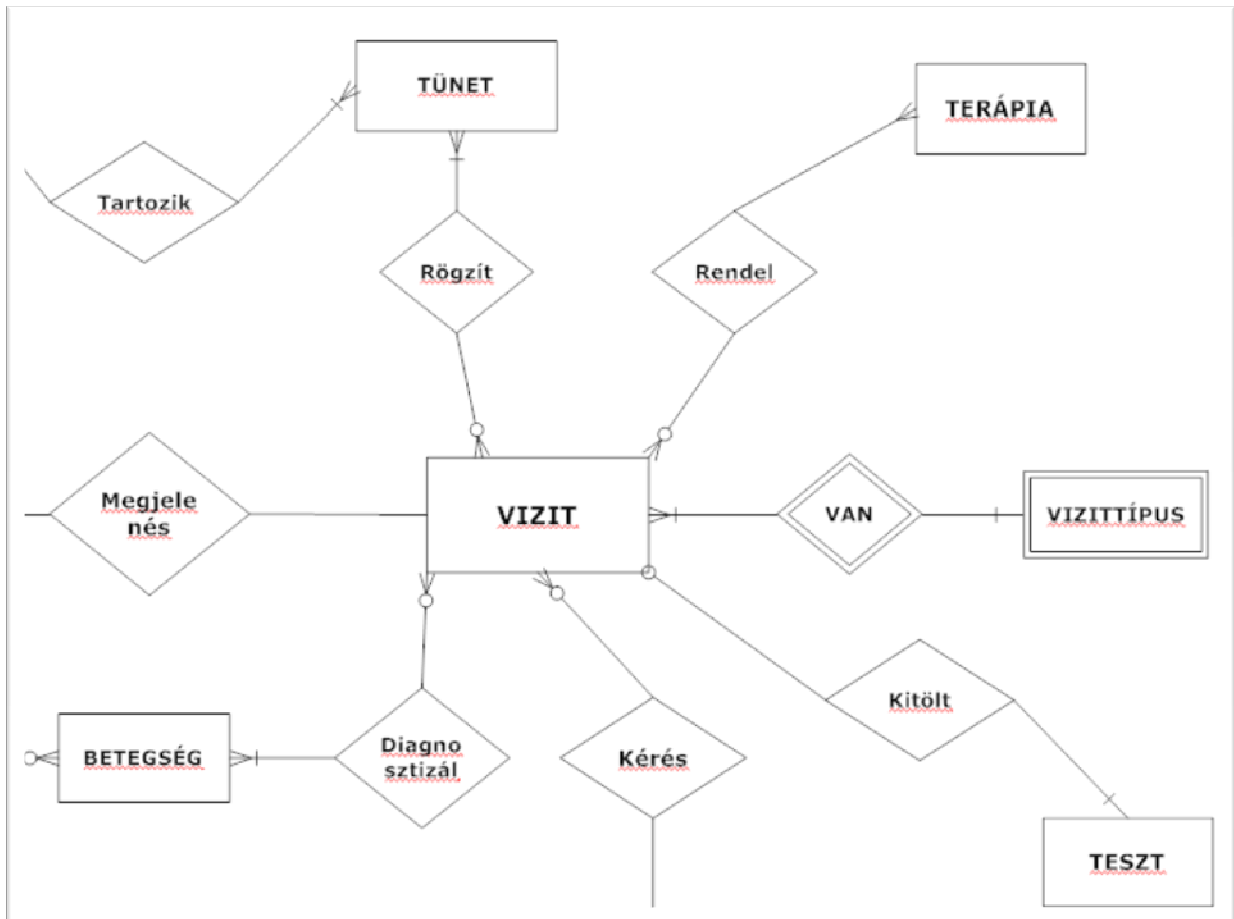
Mivel a myositis lefolyása betegségtípusonként és egyénenként nagymértékben különböző lehet, a kimenetelét is nehéz általánosságban megjósolni. Az biztosan elmondható, hogy az orvosi utasítások pontos betartása rengeteget segít az állapot gyors és tartós javulásában. A halálos kimenetel szerencsére ritka, de egyéb társbetegségek vagy kifejezetten súlyos, a terápiára nem reagáló myositis a beteg halálához vezethet.

11.2 ER modell



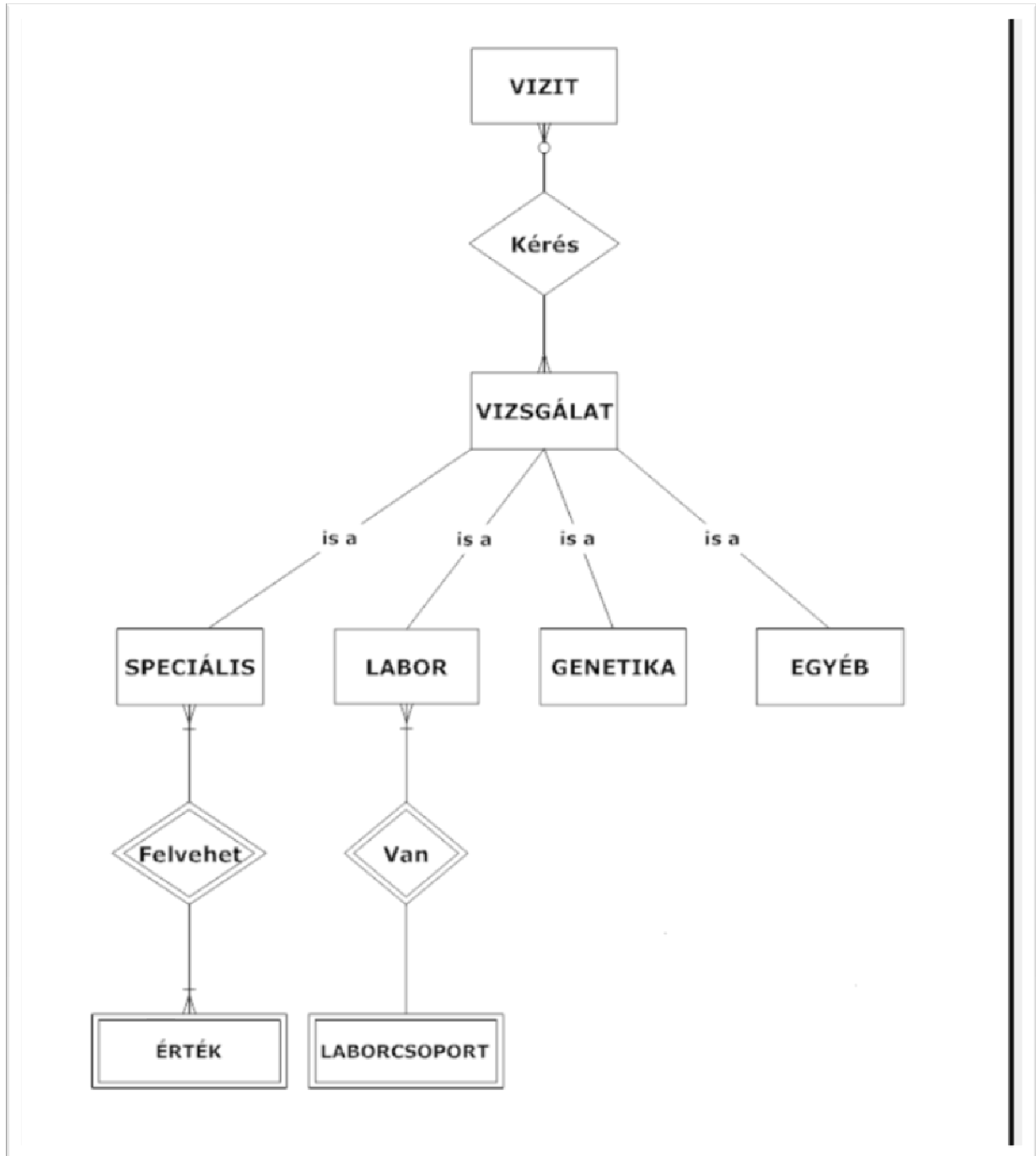
34. ábra: ER modell (1. rész)

ER modell



35. ábra: ER modell (2. rész)

ER modell



36. ábra: ER modell (3. rész)

11.3 Ábrajegyzék

1. ábra: ADAT - adatbázis - INFORMÁCIÓ 5

2. ábra: A hagyaték adatbázis táblái.....	13
3. ábra: 1Névsor tábla.....	13
4. ábra: 2Klinikai adatok	14
5. ábra: 3Kritérium tünetek	15
6. ábra: 4tünetek tábla	16
7. ábra: 5Börtünetek tábla	16
8. ábra: 7Anamnézis tábla	17
9. ábra: H pylori titer tábla.....	18
10. ábra: Képkalkotó tábla.....	19
11. ábra: Terápia tábla.....	19
12. ábra: Myositis Egyed típus lista	25
13. ábra: Myositis Kapcsolat típus lista.....	25
14. ábra: Myositis adatbázis relációséma (1. rész)	28
15. ábra: Myositis adatbázis relációséma (2. rész)	29
16. ábra: Myositis adatbázis relációséma (3. rész)	30
17. ábra: Myositis adatbázis megvalósítás	33
18. ábra: Myositis relációs nézete	34
19. ábra: Migráció / Adatimport.....	37
20. ábra: Lekérdezés 1.	39
21. ábra: Eredménytábla 1.	39
22. ábra: Lekérdezés 2.	41
23. ábra: Eredménytábla 2.	41
24. ábra: Lekérdezés 3.	42
25. ábra: Eredménytábla 3.	43
26. ábra: Lekérdezés 4.	44
27. ábra: Eredménytábla 4.	44
28. ábra: SmartDraw indítóképernyő.....	45
29. ábra: SmartDraw ER diagram.....	46
30. ábra: XAMPP Control Panel	48
31. ábra: phpMyAdmin.....	49
32. ábra: phpMyAdmin adatbázis beállítások.....	49
33. ábra: Myositis adatbázis phpMyAdminban	50
34. ábra: ER modell (1. rész).....	56
35. ábra: ER modell (2. rész).....	57
36. ábra: ER modell (3. rész).....	58

11.4 Etikai nyilatkozat

Debreceni Egyetem
Orvos és Egészségtudományi Centrum
Belgyógyászati Intézet
III. sz. Belgyógyászati Klinika
Igazgató: Prof. Dr. Zóher Margit
4004 Debrecen, Móricz Zs. u. 22
Levelelm: 4004 Debrecen, Pf. 3.
Telefon: Igazgató: 52/414-969 Fax: 52/414-969,
posta: 52/453-337



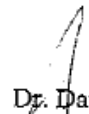
University of Debrecen
Medical and Health Science Center
Institute for Internal Medicine
3rd Department of Medicine
Director: Professor Margit Zóher M.D., Ph.D., D.Sc.
Móricz Zs. Str. 22. Debrecen 4004
4004 Debrecen P.O. Box 3 Hungary
Tel: Director: (36-52) 414-969 Fax: (36-52) 414-969
Reception: (36-52) 453-337

Nyilatkozat

Tárgy: Etikai szabályok betartása

Alulírott Dr. Dankó Katalin, a debreceni Myositis Munkacsoport vezetője igazolom, hogy a Hídi Viktória által készített myositis adatbázis a betegeknek tudtával és beleegyezésével készült. Munkánk során mind az országos, mind a Regionális és Intézményi Kutatásetikai Bizottság által hozott szabályokat, valamint a 23/2002. (V. 9.) EüM rendelet ("az emberen végzett orvostudományi kutatásokról") ide vonatkozó rendelkezéseit betartottuk.

Debrecen, 2010 április 12.


Dr. Dankó Katalin
Mcd. Habil., egyetemi docens
MTA doktora