

Telemedicina-fejlesztés: Az adatminőség modellezése és vizsgálata telemedicina rendszerekben és az új tervezési minták produktivitásra gyakorolt hatása a modern webalkalmazás-fejlesztésben

PHD ÉRTEKEZÉS TÉZISEI

Jánki Zoltán Richárd

Témavezető:

Dr. Bilicki Vilmos, PhD
egyetemi adjunktus

Szegedi Tudományegyetem
Természettudományi és Informatikai Kar
Informatika Doktori Iskola
Szoftverfejlesztés Tanszék

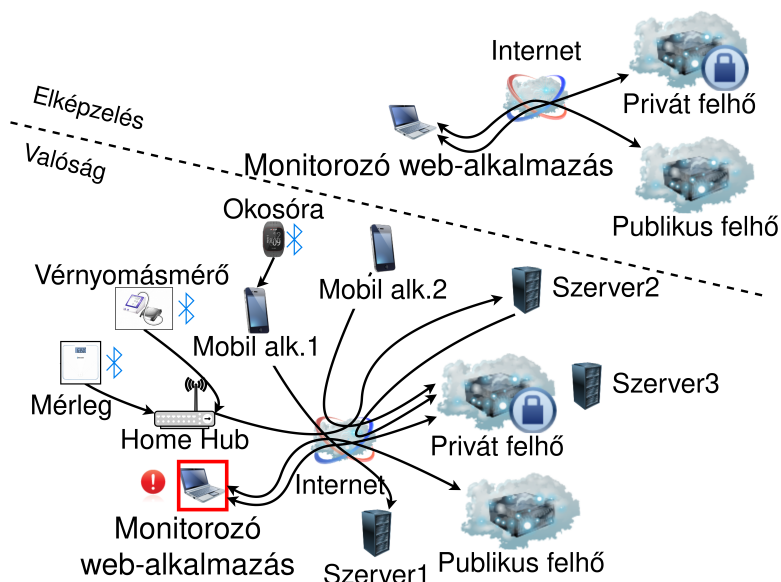


Szeged
2023

1. Bevezető

A telemedicina napjaink egyik leggyorsabban fejlődő területe, melyet egyre több szoftveralkalmazás támogat. A mai telemedicina rendszerek már ritkán épülnek egyszerű klienszerver architektúrára, inkább a bonyolult adatutak a jellemzőek. Továbbá megfigyelhető, hogy a mikroszolgáltatás-szintű megoldások és a már meglévő, jól működő megoldások integrálása egyre népszerűbb. Tapasztalataink során mi is ezzel szembesültünk, és létfontosságú, hogy ne csak a telemedicinát támogató szoftverekre helyezzük a hangsúlyt, hanem azokra az eszközökre és módszerekre is, amelyek megkönnyítik a telemedicina szoftverek fejlesztését és könnyen alkalmazkodnak más területek szoftverfejlesztési folyamataihoz is.

A doktori értekezés a telemedicina rendszerfejlesztés főbb kihívásait mutatja be az adatminőség és a fejlesztői produktivitás szempontjából. Bemutattunk számos modellezési, szimulációs, gráf-alapú és közelítő módszert, amelyekkel a komplex telemedicina rendszerek adatminősége és konzisztenciája mérhető. A WebDAO tervezési minta alkalmazásával egy saját fejlesztésű SDK-t mutattunk be, amely nem csak javítja az adatintegritást, hanem növeli a produktivitást is. A fejlesztéshez szükséges idő mérésén túl az eredményeket repository metrikák összehasonlításával is validáltuk. Tapasztalataink alapján arra a következtetésre jutottunk, hogy a strukturális és architektúrális tervezési minták nagyon fontos szerepet játszanak és használatuk rengeteg előnyt von maga után. Ennek eredményeként különféle természetes nyelvfeldolgozási, úgynevezett NLP technikát dolgoztunk ki annak érdekében, hogy még jobban megértsük az alkalmazott legjobb gyakorlatokat, és mi magunk is javaslatokat tehesünk a modern webfejlesztés gyakorlatára.



1. ábra. A telemedicina rendszerek elképzelt és valós struktúrája

2. I. téziscsoport: Adatkonzisztencia és adatminőség elemzése elosztott telemedicina rendszerekben

Egy modellező és szimulációs környezetet készítettem, amellyel elosztott telemedicina rendszerek adatminőségét és adatkonzisztenciáját mérhetjük. A szimulációs eredmények megmutatták, hogy a késleltetés jelentősen befolyásolhatja a konzisztenciát és az adatminőséget. Az eredmények alapján egy taxonómiát dolgoztam ki, amely az elosztott telemedicina rendszerekben jelenlévő kompromisszum problémára nyújt egy megoldást. Emellett egy ilyen rendszer formális specifikációját is elkészítettem. További vizsgálatokkal megmutattam, hogy egy gráf-alapú reprezentációval hatékonyan kiértékelhetőek az elosztott rendszerek. A DAG struktúrát kihasználva különböző gráf algoritmusokkal együtt a rendszert leíró állapotter komplexitása jelentősen egyszerűsödik. Végül egy keretrendszert készítettem el, a TSMOF-et, amellyel az adatminőséget leíró állapotter egy fókuszált részét vizsgálhatjuk a rendszer testreszabható működése mellett. A keretrendszer emulációs, szimulációs, modellező és közelítő módszereket alkalmaz.

A téziscsoporthoz kapcsolódó publikációk: [J1], [J2], [J3], [FR2], [FR3]

Egy telemedicina rendszer számos komponensből épül fel, amely könnyedén komplex adatutakhoz vezethet. Az adat ezen bonyolult út során számos olyan transzformációs, aggregációs és leképezési folyamaton megy keresztül, amelyek során az adat sérülhet. Az adat sérülését gyakran késleltetés vagy helytelen rendszerkonfiguráció okozza, így fontos, hogy megtaláljuk a használati esethez legjobban illeszkedő beállításokat. Ez a téziscsoport egy taxonómiát mutat be a telemedicina használati esetekről és a javasolt rendszerkonfigurációkról. Az osztályozás a CAP és PACELC tételek mentén lett kidolgozva. A taxonómia validálása egy ilyen telemedicina rendszert leíró formális specifikációval és az ezen specifikáción végzett adatminőség-szimulációval történt meg. A későbbiekben bemutatásra kerül egy gráf-alapú megközelítés a szimulációs eredmények hatékony kiértékelésére. Megfigyelhető, hogy az adatminőséget leíró állapotter óriási méreteket is ölthet, így ésszerű annak egy fókuszált részét vizsgálni csak. Így végül bemutatásra kerül egy közelítő módszerrel alapuló technika az adatminőséget leíró állapotter komplexitásának csökkentésére, amellyel pontos eredményeket adhatunk valós idejű telemedicina rendszerek adatminőségére valós algoritmus futtatások mentén.

2.1. I/1. tézis: Elosztott telemedicina rendszerek taxonómiája a CAP és PACELC tételek mentén

Bemutattam az elosztott telemedicina rendszerek kompromisszum problémáját a CAP tétel mentén. Egy formális módszert adtam az ilyen jellegű rendszerek modellezésére és szimulációjára. Telemedicina használati esetek mentén azonosítottam a kompromisszum problémájának a létezését, és kiértékeltem a késleltetés adatkonzisztenciára és adatminőségre gyakorolt hatását. Javasoltam egy taxonómiát az elosztott telemedicina rendszerekre figyelembe véve a konzisztencia és elérhetőség közötti kompromisszum kérdését.

A tézishöz kapcsolódó publikációk: [J1], [FR2], [FR3]

Erich Brewer CAP tétele alapján [3] nem létezik olyan elosztott rendszer, amely a három alapvető képesség közül egyidejűleg kettőnél többet tudna garantálni. Ezek a konzisztencia (C), elérhetőség (A) és particionálás-tűrés képességek. A korlát miatt egy úgynevezett kompromisszumot kell kötni, és ez megfogalmazta a PACELC tételt [1]. A tétel szerint, hálózati particionálás (P) esetén az elérhetőség (A) és a konzisztencia (C) között kell kompromisszumot kötni, minden más esetben – ha a rendszer normálisan működik, particiók nélkül – a kompromisszumot a késleltetés (L) és a konzisztencia (C) között kell kötni. A telemedicina rendszerek széles skálája miatt különösen fontos, hogy a megfelelő egyensúly megtaláljuk függően a használati esettől. Az egyensúly megtalálása végett megvizsgáltuk az adatot a konzisztencia függvényében. A Peter Bailis és társai [2] által bemutatott ötlet mentén az adathoz hozzárendelhetünk egy k -romlási paramétert, amely leírja, hogy legfeljebb k -verzióval régebbi adatot fogadunk el. Egy jól paramétrezhető rendszer modellezését követően a szimulációs eredmények megmutatták, ha a k értéket 0-ra állítjuk, közel 100%-os konzisztenciát érhetünk el, de a késleltetés ezt jelentősen ronthatja. A konzisztencia nagyságrendileg 5%-kal csökken a k érték 1-gyel történő növelésével, illetve a késleltetés egy újabb 5%-kal ronthatja az adatkonzisztenciát 100 ms késleltetéstöbblet esetén. Az eredmények alapján megalkottunk egy taxonómiát a kompromisszum probléma kezelésére elosztott telemedicina rendszerek esetén, amelyet az 1. táblázat mutat. A használati eseteket 3 kategóriába soroltuk: nem-offline, fél-offline és offline telemedicina, figyelembe véve, hogy a gyorsítótárazás megengedett-e. Megfigyelhető, hogy az adatromlás a gyorsítótárazástól függetlenül bekövetkezhet. Ha a gyorsítótárazást nem engedjük meg, javasolt a PC/EL rendszer használata. Ha a gyorsítótár használata megengedett, akkor érdemes a konzisztenciára fókuszálni, de alacsony k -romlási paraméterrel. Az offline telemedicina esetében az elérhetőség fontosabb, mint a konzisztencia, ezért a magasabb k -romlási paraméter használata is megengedett.

1. táblázat. *Telemedicina rendszerek taxonómiája a kompromisszum probléma függvényében*

Használati eset kategória	Adatromlás valószínűsége	Javasolt rendszer
Nem-offline telemedicina	Nem	PC/EL gyorsítótár nélkül
	Igen	PC/EC gyorsítótár nélkül
Fél-offline telemedicina	Nem	PC/EL gyorsítótárral magas k -romlási paraméter
	Igen	PC/EL gyorsítótárral alacsony k -romlási paraméter
Offline telemedicina	Nem	PA/EL gyorsítótárral magas k -romlási paraméter
	Igen	PA/EL gyorsítótárral magas k -staleness paraméter

2.2. I/2. tézis: Gráf-alapú módszer az adatminőség mérésére telemedicina rendszerekben

Bemutattam – az I/1. tézisben definiált formális módszerrel, – hogy az elosztott telemedicina rendszerek adatminőségét leíró állapottere leírható egy olyan gráffal, amelynek DAG struktúrája van. A struktúra előnyeit kihasználva, gráf klaszterezést hajtottam végre a gráf gyengén összefüggő komponensein és optimalizáltam az adatminőséget leíró állapotter komplexitását.

A tézishez kapcsolódó publikációk: [J2]

A szimuláció egy széles körben elfogadott és alkalmazott technika rendszerek adott körülmények között történő viselkedésére [12]. Leslie Lamport bemutatta a Temporal Logic of Actions (TLA) módszert, amellyel konkurens rendszereket specifikálhatunk [9]. TLA-ban gyűjtött tapasztalataink után modelleztünk több telemedicina használati esetet, és megvizsgáltuk, hogy a konzisztencia és az adatminőség hogyan változik a gyorsítótárazás és a késleltetés szempontjából. A TLA-ban elkészült rendszermodell helyessége a TLA+ Model Checker-rel (TLC-vel) ellenőrizhető [14]. A TLC kimeneti gráfja magába foglalja a rendszer összes lehetséges állapotát. Megfigyeltük, hogy egy relatív kicsi rendszer is milliónyi pontot és élt tartalmazhat, egy a teljes gráf kiértékelése szinte lehetetlen. A gráfok struktúrájának elemzése során megfigyeltük, hogy a gráfok tartalmaznak köröket, azonban kizárólag a leveleknél vannak saját magukba mutató élek. Az úgynevezett termináló élek eltüntetésével viszont a gráf irányított körmentes lesz, azaz DAG struktúrát visel. A DAG struktúra tudatában azonban a szimulációs analízis jelentősen egyszerűsödik köszönhetően a gráf algoritmusoknak. A gráfban minden szimulációs futtatást egy gyengén összefüggő komponens reprezentál. A gráf gyengén összefüggő komponensein végrehajtottunk egy klaszterezést a levelekben található adatminőség értékek alapján, és megfigyeltük, hogy az azonos adatminőséggel bíró komponenseknek azonos gráf struktúrája van. A 2. táblázat mutatja, hogy a klaszterezéssel csupán az adatminőséget leíró állapotter 2%-át kell vizsgálnunk. Azonban fontos megemlíteni, hogy vannak olyan klaszterek, amelyek csak 1-2 komponenst tartalmaznak, de vannak olyanok is, amelyek több ezret. A DAG struktúra előnyeit kihasználva a gráf algoritmusok futási ideje már lineáris. Összefoglalva, az adatminőséget leíró állapotter komplexitása jelentősen csökken, ha a gráf-alapú megközelítést és a klaszterezést együtt alkalmazzuk.

2. táblázat. A gyengén összefüggő komponensek száma és a k -romlási paraméterértékekkel generált klaszterek száma

k -romlási paraméter	A modell ellenőrzés során létrejött komponensek száma	Az adatminőség alapú klaszterezés során létrejött klaszterek száma
0	276.862	5.383
1	288.988	4.735
2	282.104	4.112
3	282.032	4.139

2.3. I/3. tézis: Elosztott telemedicina rendszerek modellezése, emulációja és szimulációja közelítő módszerekkel

Közelítő módszereket javasoltam, hogy megtaláljuk azokat a legjobban illeszkedő valószínűségi sűrűségfüggvényeket, amelyek egy komplex rendszer folyamatait írják le. Elkészítettem egy keretrendszert, amely az adatminőséget leíró állapottér egy fókuszált részét vizsgálja a rendszer testreszabható működése esetén. A keretrendszer hatékonyságát bemutattam valós algoritmusok futtatásán keresztül. Kimutattam, hogy jelentős adatminőségromlás tapasztalható a késleltetés miatt valós idejű telemedicina rendszereknél.

A tézishez kapcsolódó publikációk: [J3]

A telemedicinában az adatút rendkívül komplex lehet, amely tartalmaz számos adatmanipulációs, aggregációs fázist is. Megfigyeltük, hogy a késleltetés komoly hatást gyakorolhat az adatminőségre, ami valós idejű rendszerek esetében még kritikusabb lehet. A valós-idejű telemedicina használati esetünk négy egymástól függő folyamatot tartalmaz: elektrokardiogram (EKG) mérést, adatfeltöltést, pitvarfibrilláció (PF) detektálást és PF osztályozást. Ha a folyamatok sokáig tartanak, a rendszer már nem marad valós idejű, és értékes adatokat kell eldobnunk, hogy fenntartsuk a valós idejű képességét, ami viszont adatminőség-csökkenést eredményez. Az adatminőségromlás mértékének megfigyelésére, modellező és szimulációs módszereket lehet alkalmazni. Modelleztünk egy valós idejű EKG rendszert, amely tartalmaz egy PF detektáló és PF osztályozó algoritmust. A modell az [1.](#) kódnál látható. Habár a rendszert logikailag modellezzük, az adatminőséget leíró állapottér még mindig óriási lehet egy kis rendszer esetén is, amelynek kiértékelése évekbe telhet.

1. kód. Egy valós idejű telemedicina rendszerben a késleltetés formális modellje

```
CONSTANT MAX_NUM_WIN, WINDOW_SIZE, TIMEOUT_UP
VARIABLES numWin, now1, now2, now3, now4, dropList, ui, pi1, pi2

Step(t) == /\ IF numWin[t] < MAX_NUM_WIN
            THEN /\ numWin' = [numWin EXCEPT ![t] = numWin[t] + 1]
                 /\ now1' = [now1 EXCEPT ![t] = now1[t] + WINDOW_SIZE]
                 /\ now2' = [now2 EXCEPT ![t] = now1'[t] + ui[t]]
                 /\ IF now3[t] < now2'[t] + TIMEOUT_UP
                    THEN /\ dropList' = [dropList EXCEPT ![t] = <<FALSE>> \o
                        dropList[t]]
                        /\ now3' = [now3 EXCEPT ![t] = now2'[t] + pi1[t]]
                        /\ IF now3'[t] > now4[t]
                           THEN /\ now4' = [now4 EXCEPT ![t] = now3'[t] +
                               pi2[t]]
                               ELSE /\ UNCHANGED <<now4>>
                        ELSE /\ dropList' = [dropList EXCEPT ![t] = <<TRUE>> \o
                            dropList[t]]
                            /\ UNCHANGED <<now3, now4>>
                 /\ ui' = [ui EXCEPT ![t] = (UPLOAD_LATENCY[numOp[t] + 1)]
                 /\ pi1' = [pi1 EXCEPT ![t] = (PROC_TIME_ALG_1[numOp[t] +
```

```
1]])  
/\ pi2' = [pi2 EXCEPT ![t] = (PROC_TIME_ALG_2[numOp[t] +  
1]])  
ELSE /\ UNCHANGED vars
```

Ha a rendszer vagy egy folyamat implementációja elérhető, annak viselkedését könnyen emulálhatjuk virtualizációval, és különböző konfigurációk mellett tesztelhetjük annak működését. Az emulációs eredmények alapján láthatjuk, hogy egy folyamat vagy a rendszer milyen késleltetéssel működik. Az emulációs eredmények és közelítő módszerek segítségével jól meghatározható a folyamat késleltetését leíró valószínűségi sűrűségfüggvény. A sűrűségfüggvényt használva generált véletlen értékek tükrözik a folyamat valószínűsítését. Ha minden kritikus komponens és folyamat emulálható vagy közelíthető ily módon, akkor pontosabb adatminőségi értékeket számíthatunk ki a teljes rendszerre a modell segítségével, és kiértékelhetjük az adatminőséget leíró állapotteret fókuszáltabb részét. A kiválasztott használati esetre alkalmazva a módszert megfigyelhető, hogy az adatminőség romlása akár 4,25%-os is lehet az első aggregáció után, de a második aggregáció során ez a mérték már akár 20%-os is lehet.

3. II. téziscsoport: Produktivitást növelő szoftverfejlesztői csomag hibrid felhő támogatással

Egy új tervezési mintát mutattam be, a WebDAO-t, illetve jelentőségét a modern front-end alkalmazásfejlesztésben. Megmutattam, hogy az FHIR egy széles körben elfogadott szabvány, és ennek egy SDK-ba öntött implementációját is elkészítettem hibrid felhő támogatással. Kimutattam, hogy a WebDAO-t alkalmazó szabványos, telepíthető SDK javítja a fejlesztői produktivitást. A vizsgálatot kiterjesztve nyíltforrású projektekre, repository metrikák mentén bemutattam, hogy a WebDAO javítja a fejlesztői produktivitást.

A téziscsoporthoz kapcsolódó publikációk: [J4], [J6], [C1], [FR1], [FR4]

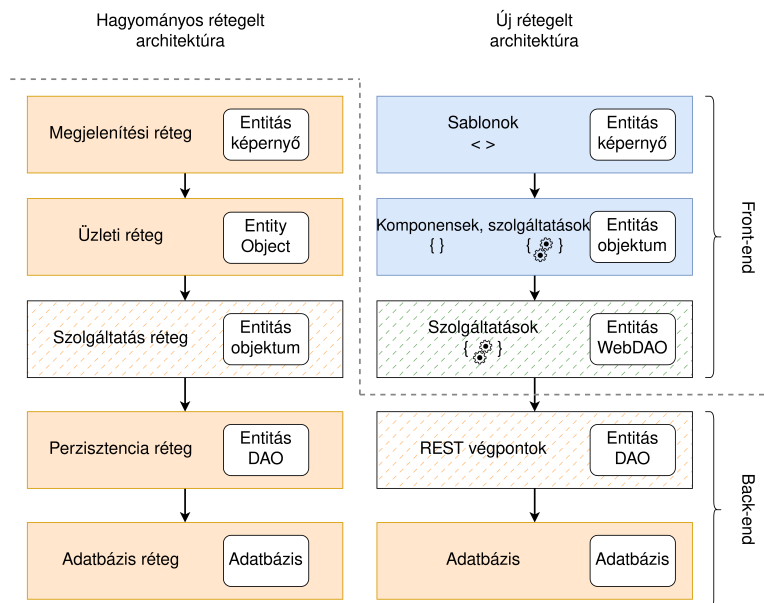
A szoftverfejlesztésben gyakran szembesülünk visszatérő vagy hasonló problémákkal, kihívásokkal. A gyakorta visszatérő hibákra adott megoldások idővel legjobb gyakorlatokká fejlődnek, majd a legjobb gyakorlatok általánosításával keletkeznek a tervezési minták [15]. A tervezési minták nem csak problémák megoldására alkalmasak általános-ságban, hanem számos pozitív hatást visznek magukkal a szoftverfejlesztési folyamatokba. A kód újrahaználhatósága és fenntarthatósága fontos szempontok [6], de a produktívítás növelése sem elhanyagolható. Ebben a téziscsoportban bemutatunk egy SDK-t, amely nemcsak standardizálja a front-end és a back-end kapcsolatát a modern webfejlesztésben, hanem hibrid felhő támogatást nyújt, és jelentősen növeli a fejlesztői produktivitást a WebDAO tervezési minta alkalmazásával. Végül bemutatjuk a WebDAO fontosságát több ezer nyíltforrású Angular projekt elemzésén keresztül.

3.1. II/1. tézis: Szabványos front-end és back-end kapcsolat hibrid felhő támogatással

Bemutattam, hogy a front-end és back-end kapcsolata szabványosítható a WebDAO tervezési mintával a modern webfejlesztésben. Különösen igaz ez a telemedicinára, ugyanis az FHIR szabvány és a WebDAO tökéletes kombinációt alkot. Használati statisztikákat készítettem az FHIR-ről a webfejlesztésben. Bemutattam, hogy a rétegelt architektúra hogyan változott meg a szervermentes fejlesztés miatt, és példán keresztül felvázoltam az új architektúra használatát az Angular alkalmazásfejlesztésben.

A tézishez kapcsolódó publikációk: [J4], [J6]

A szoftverfejlesztésben fontos, hogy egy szoftver tervezése rendszer szinten is megtörténjen, és definiáljuk az architektúrát, beleértve az egyes komponensek felelősségét. A modern webfejlesztésben az úgynevezett rétegelt architektúra vagy n-szintű architektúra az egyik leggyakrabban alkalmazott architekturális tervezési minta, amely ma már egy de facto szabvány is [11]. A hagyományos rétegelt architektúra öt rétegből áll – amelyekből egy opcionális – jól definiált szerepkörökkel. Napjainkban a szervermentes fejlesztés egyre nagyobb teret nyer, ezzel a backend elvékonyodik, és egyre több felelősség kerül át a front-end-re. Az Angular alkalmazásfejlesztésben a szolgáltatások képezik a szolgáltatási réteget a hagyományos rétegelt architektúrában, viszont ezzel a megközelítéssel a felső három réteg elszakad az alsó kettőtől. Annak érdekében, hogy a front-end és a back-end közötti kapcsolatot fenn tudjuk tartani, javasolt bevezetni egy széles körben elismert szabványt, mint például a Fast Healthcare Interoperability Resources-t (FHIR-t) [4]. Ez a tézispont bemutatja, hogy az FHIR a leggyakrabban alkalmazott szabvány egészségügyi rendszereknél, és nyíltforrású projektek ezrei használják világszerte. A standardizált domain modellen túl ajánlott a Data Access Object (DAO) képességeket átmozgatni a front-end-re. Ahogy azt a 2. ábra is mutatja, erre a tervezési mintára WebDAO-ként hivatkozunk az új rétegelt architektúrában. A telemedicina projektek fejlesztése során megfigyeltük, hogy a WebDAO kiemelt szerepet játszik, ugyanis nem csak szabványosítja a kapcsolatot a front-end és a back-end között, de számos kiegészítő szolgáltatás adható hozzá a projektekhez.



2. ábra. A hagyományos és az új rétegelt architektúra

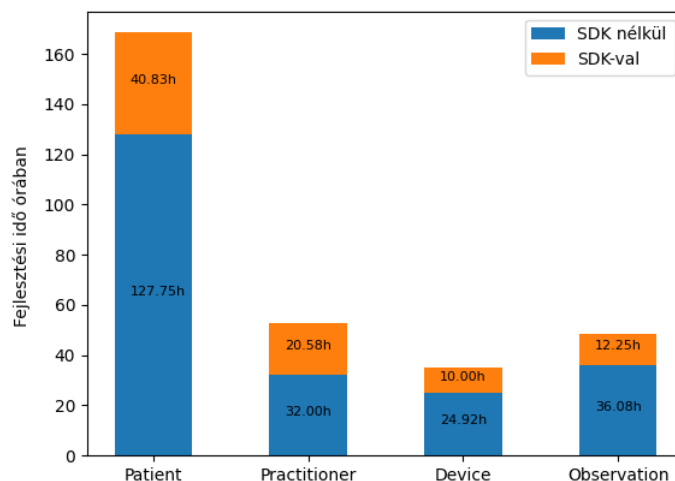
3.2. II/2. tézis: A WebDAO alkalmazása és produktivitás mérése a telemedicina alkalmazásfejlesztésben

Lefejlesztettem egy SDK-t a telemedicina alkalmazások fejlesztéséhez, amely hibrid felhő támogat és a WebDAO tervezési mintát alkalmazza. Javasoltam egy módszertant, amely bemutatja, hogy a produktivitás számszerűsíthetően javul a WebDAO tervezési minta használata mellett. Kiértékeltem a fejlesztői produktivitást valós fejlesztők bevonásával, és kimutattam, hogy a WebDAO-val a fejlesztői produktivitás jelentősen javítható.

A tézishez kapcsolódó publikációk: [J6], [C1], [FR1], [FR4]

A tervezési mintáknak számos hatásuk lehet a szoftverfejlesztési folyamatokra. Lefejlesztettük az Included SDK-t [7], amely magán viseli a WebDAO képességeit, a hibrid felhő támogatást, a polyglot perzisztenciát és egy standardizált domain modellt kínál az FHIR-rel. Az SDK-nk már számos telemedicina pilot projekt fejlesztése alatt jól teljesített. Azt tartják, hogy a DAO javítja a fejlesztői produktivitást a kód újrahasználatosságának és az implementáció egyszerűségének köszönhetően. Mivel a WebDAO hasonló képességekkel bír, csak át lett mozgatva a front-end-re, a hipotézisünk az, hogy a WebDAO pozitív hatással van a produktivitásra. További kutatási kérdés, hogy ha a WebDAO pozitívan hat a produktivitásra, akkor ez számszakilag milyen mértékű? Annak érdekében, hogy a WebDAO hatásait demonstráljuk, egyetemi hallgatókat vontunk be, akik informatika képzésben vesznek részt, és az őik produktivitását mértük. Feladatuk volt Angular szolgáltatások fejlesztése a négy legfontosabb FHIR erőforráshoz a telemedicina pilot projektjeinkben az Included SDK-t használva és anélkül. Az Included SDK tartalmazza a létrehozás, olvasás, frissítés és törlés (CRUD) műveleteket az FHIR erőforrásokhoz, az erőforrásokhoz ajánlott szűrési képességeket és a szükséges FHIR interfészeket. A 3. ábrán látható, hogy a fejlesztési idő legalább 50%-kal csökkent, ha a SDK-t és ezzel együtt a WebDAO-t alkal-

mázták az Angular szolgáltatások fejlesztéséhez. Habár a bevont hallgatók száma meglehetősen kicsi, a DAO és WebDAO produktivitásra gyakorolt pozitív hatását számszerűsítve is igazoltuk.



3. ábra. Átlagos fejlesztési idő az SDK használatával és használata nélkül

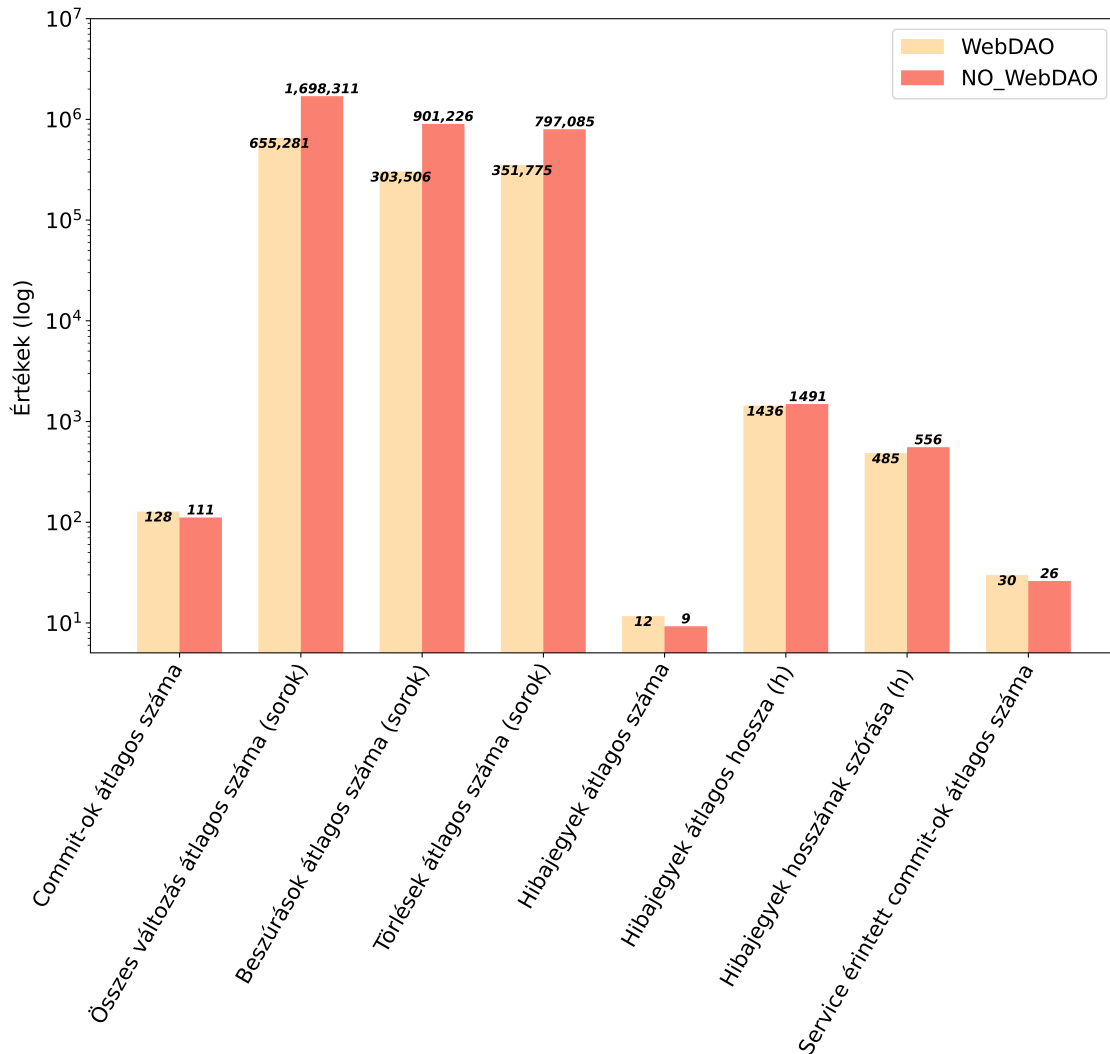
3.3. II/3. tézis: Repository metrikákon alapuló produktivitásmérés és a WebDAO széleskörű alkalmazása

Bemutattam egy repository metrikákon alapuló mérési módszert a fejlesztői produktivitásra. Az eredmények alapján kimutattam, hogy a WebDAO Angular alkalmazásfejlesztésben történő használatával a forráskód változások jelentősen csökkenthetők, és a hibajegyek megoldása kevesebb időt igényel.

A tézishez kapcsolódó publikációk: [J4]

Mivel a WebDAO tervezési mintát még eddig senki sem mutatta be ilyen formában, de a gyakorlatban jelentős előrelépésnek tartjuk, feltételeztük, hogy még ha nem is tudatosan, talán mégis alkalmazzák a modern webfejlesztésben, különösen Angular projektekben. Ha a feltételezésünk helyes, akkor a produktivitást más aspektusokból is mérhetjük [8, 10, 13]. A validációhoz letöltöttünk 19.116 nyíltforrású Angular projektet, és megvizsgáltuk az implementációt a III. téziscsoportban és a III/1. tézispontban bemutatott módszerekkel. Megfigyeltük, hogy a legtöbb fejlesztő a szolgáltatásokat a back-enddel létrejövő kapcsolat kiépítésére használja, ami természetesen a legjobb gyakorlat, de ezek jelentős része WebDAO-ként is funkcionál. Összehasonlítottuk a projektekben elért produktivitást a repository metrikák mentén. A 4. ábrán látható, hogy a commit-ok és a hibajegyek száma igen hasonló mindkét esetben. Azonban a forráskód módosítások száma fele annyi, ha alkalmazzuk a WebDAO-t, ami azt indukálja, hogy ezeknél a projekteknél gondosabban kivitelezett tervezési fázis előzte meg a fejlesztést, és kevésbé jellemző a vízésés-modellszerű fejlesztés. Habár a hibajegyek száma egészen hasonló, megfigyelhető,

hogy a hibajegyek átlagos hossza és azok szórása jelentősen alacsonyabb, ami arra enged következtetni, hogy a hibajegyek hamar megoldásra kerülnek, és a problémamegoldás sokkal fókuszáltabb és hatékonyabb a WebDAO használata esetében. Összességében, a WebDAO egy implicit alkalmazott tervezési minta az Angular alkalmazásfejlesztésben, és pozitív hatása van a produktivitásra nézve a repository metrikák alapján.



4. ábra. Projekt metrikák a WebDAO használatának függvényében

4. III. téziscsoport: Tervezési minták azonosítása természetes nyelvfeldolgozási technikákkal

A WebDAO bemutatását követően készítettem egy Véletlen erdő (Random Forest) osztályozót és egy BERT modellt, amellyel a tervezési mintát azonosíthatjuk nyíltforrású projektekben. Megmutattam, hogy a WebDAO-t implicit is alkalmazzák az Angular projektekben. Meg-

figyeltem, hogy több Model-View-alapú tervezési mintát is alkalmaznak a modern webfejlesztésben, így kidolgoztam egy taxonómiát ezekre, és használati eseteket rendeltem hozzájuk. Végül készítettem egy szabályrendszert, aminek segítségével az MVW tervezési minták azonosíthatóak nagy pontossággal Angular projektekben nagy nyelvi modellekkel.

A téziscsoporthoz kapcsolódó publikációk: [J4], [J5]

A tervezési minták esszenciális építőkövek a szoftverfejlesztésben, ugyanis olyan általánosan alkalmazható legjobb gyakorlatokat nyújtanak a fejlesztőknek, amellyel sokkal fenntarthatóbb, skálázhatóbb és érthetőbb szoftvert készíthetnek. Ma már számos architektúráis, strukturális és viselkedési tervezési minta ismert [5, 6], melyek alapelvei ugyan jól definiáltak, mégis nagyban függ a rendelkezésre álló technológiától és keretrendszerrel az alkalmazásuk, struktúrájuk és szerepköreik. A 3. és 4. fejezetre hivatkozva a szoftvereket rendszer szinten vizsgáltuk és architektúráis újításokat mutattunk be. Az elért eredmények alapján hisszük, hogy a tervezési minták fontosságát népszerűsíteni kell. Egyre több kutató foglalkozik a tervezési minták azonosításával és automatizált detektálásával különböző módszereket felhasználva. A gépi tanulás és különösképp az NLP megoldások a legnépszerűbbek és leghatékonyabbak ma. Ebben a téziscsoportban bemutatjuk saját NLP megközelítéseinket a WebDAO tervezési minta Angular projektben történő magas pontosságú azonosítására, ami mellé használati statisztikákat is készítünk. Az Angular alkalmazásfejlesztésben szerzett tapasztalataink motiváltak bennünket abban, hogy megvizsgáljuk a különböző architektúráis megoldásokat az Angular projektekben. Megfigyeltük, hogy számos Model-View-alapú architektúráis tervezési minta implementálható, és mindegyiknek megvan a saját gyakorlati hozadéka is. A különbségek tisztázása végett, és hogy javaslatokat tehessünk a minták alkalmazására, egy taxonómiát dolgoztunk ki az MVW tervezési minták használatára a modern webfejlesztésben. Gyakran nehéz egyértelműen beazonosítani, hogy melyik tervezési minta került alkalmazásra, így a nagy nyelvi modellek segítségével egy új prompt engineering módszert mutattunk be a tervezési minták azonosítására.

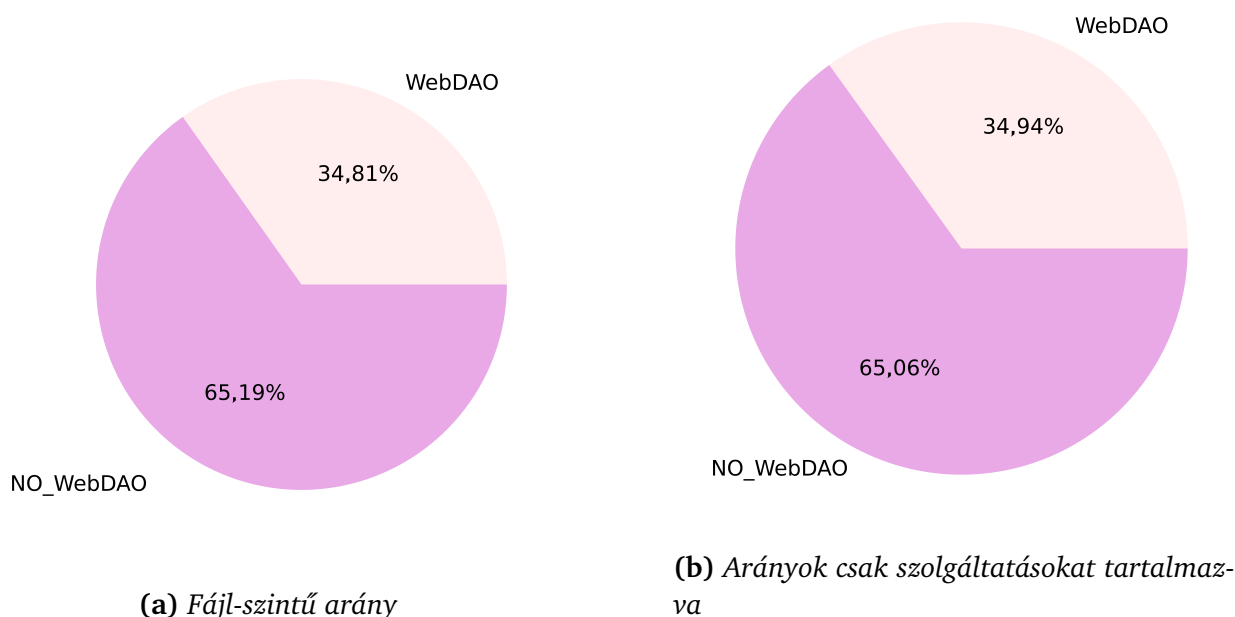
4.1. III/1. tézis: A WebDAO tervezési minta azonosítása napjaink MI technikáival

Lefejlesztettem és összehasonlítottam két NLP megoldást az Angular projektek és szolgáltatások osztályozására a WebDAO jelenlétének függvényében. Kimutattam, hogy a WebDAO egy implicit alkalmazott tervezési minta az Angular alkalmazásfejlesztésben és felfedtem, hogy számos projekt egyáltalán nem használ szolgáltatásokat.

A tézishez kapcsolódó publikációk: [J4]

A tervezési minták azonosítása segít a szoftver pontosabb megértésében, és nagyban hozzájárul a rendszer karbantartásához. A 4. fejezetben bemutattuk a WebDAO fontosságát, illetve azt, hogy gyakran alkalmazzák is Angular szolgáltatásokban. Bemutatjuk a WebDAO osztályozónkat, amely rámutat azokra az Angular szolgáltatásokra, amelyek nem felelnek meg a tervezési mintának, és kiemeli azokat a komponenseket, amelyekben a kódminőség javítható. Mivel a döntést egyetlen fájl alapján meg lehet hozni, így mi is az Angular szolgáltatásokat vizsgáltuk csak. Feltételeztük, hogy az NLP módszerek

nagyon hatékonyak lehetnek ebben az osztályozási problémában. Megcímkeztünk 504 Angular szolgáltatást, amelyből 247 a WebDAO-t implementálja, és 257-et, amely nem alkalmazza a WebDAO tervezési mintát. A szolgáltatásokat 112 érett nyíltforrású projektből gyűjtöttük ki a GitHub-ról. Elsőként egy véletlen erdő (Random Forest) osztályozót tanítottunk be, amely 89,8%-os F1-score értéket ért el 86,3%-os precision és 93,6%-os recall értékekkel. Másodikként egy Bidirectional Encoder Representations from Transformers (BERT) modellt állítottunk össze, hogy a pontosságot tovább javítsuk. A BERT osztályozónkkal már 92%-os F1-score értéket értünk mind a WebDAO és a NO_WebDAO osztályokra. Ezzel a megbízható modellel megvizsgáltunk valamennyi letöltött Angular projektet, hogy használati statisztikákat adhassunk meg a WebDAO-ra nézve. Megfigyeltük, hogy a WebDAO használati aránya közel azonos fájl szinten (lásd 5a. ábra) és projekt szinten, ahol a projekt tartalmaz szolgáltatásokat (lásd 5b. ábra). Összességében a WebDAO csupán az Angular projektek 28,09%-ában van jelen, viszont fontos megemlíteni, hogy a projektek 27%-a úgy kapott NO_WebDAO címkét, hogy egyáltalán szolgáltatás sincs benne. Ha csak azt a részhalmazát vizsgáljuk a projekteknek, amely tartalmaz szolgáltatásokat, akkor elmondhatjuk, hogy a WebDAO-t egy kicsivel több, mint a projektek 1/3-ában használják.



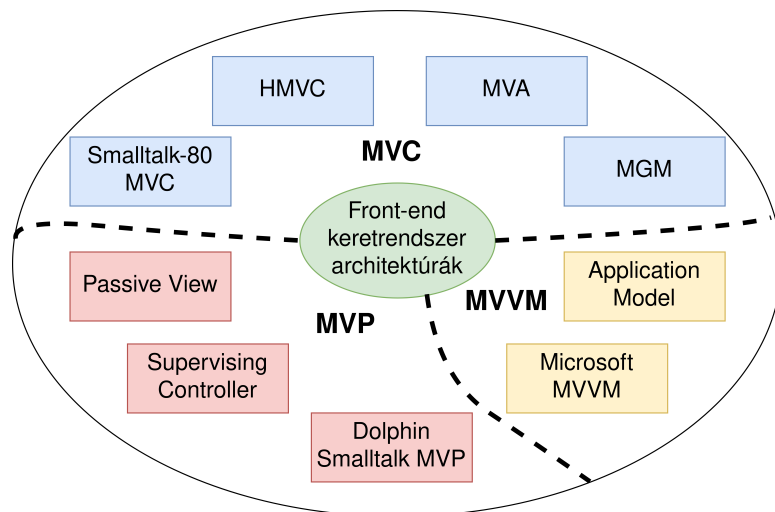
5. ábra. A WebDAO-t alkalmazó projektek aránya és a WebDAO-ként működő Angular szolgáltatások aránya

4.2. III/2. tézis: A Model-View-Whatever tervezési minta variánsainak taxonómiája a modern webfejlesztésben

Bemutattam a Model-View-alapú architektúris tervezési mintákat a modern webfejlesztésben. Javasoltam egy taxonómiát, amely az MVW tervezési minták formális definícióját és ajánlott alkalmazásait tartalmazza. Angular használati eseteket és példákat készítettem adott tervezési minták alkalmazására.

A tézishez kapcsolódó publikációk: [J5]

Számos szoftver tervezési minta ismert, amely általános legjobb gyakorlatokat kínál fenntarthatóbb és újrahasználatos forráskód érdekében. Azonban az architektúráis tervezési minták számos variánsa a különböző technológiák és keretrendszerek igényei miatt jöttek létre. Az Angular keretrendszert gyakran Model-View-ViewModel (MVVM) keretrendszernek tekintik az alkalmazott legjobb gyakorlatok miatt, azonban ezt a Google hivatalosan sosem nevezte így. Hasonlóképp, megfigyeltük, hogy a fejlesztők gyakran olyan megoldásokat alkalmaznak, amelyek ellentmondanak az MVVM elveinek. Ezért összegyűjtöttük az Angular fejlesztésben elérhető MVW tervezési minta család tagjait, amelyek könnyedén adaptálhatóak más modern keretrendszerekre. Először kategorizáltuk és felvázoltuk az elérhető tervezési mintákat (lásd [6.](#) ábra) az alapján, hogy melyik családba tartoznak. Ezt követően kidolgoztunk egy taxonómiát, amely a 9 leginkább jellemző tervezési mintát gyűjti össze és formális definíciót ad rájuk a Gamma-féle formális módszerrel. A taxonómián túl javaslatokat teszünk arra, hogy egyes tervezési minták hogyan alkalmazhatóak és a fejlesztők mikor vehetik hasznukat az Angular alkalmazásfejlesztésben.



6. ábra. Az MVW tervezési minták csoportjai és variánsai

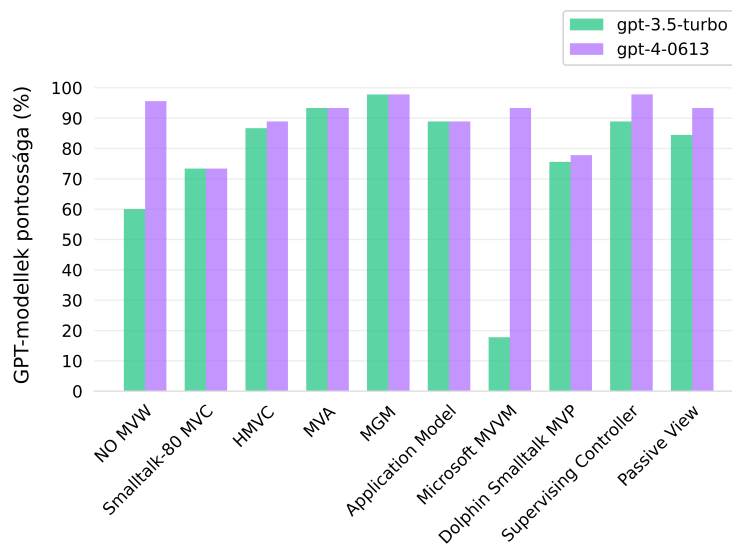
4.3. III/3. tétel: Az architektúráis tervezési minták azonosítása nagy nyelvi modellekkel

Megterveztem egy új prompt engineering módszertant tervezési minták Angular projektekben történő azonosítására. Készítettem egy szabályhalmazt, amely segít az MVW tervezési minták azonosításában Angular projektekben. Kiértékeltem mind a GPT-3.5 és GPT-4 modelleket a szabály-alapú minta azonosításban.

A tézishez kapcsolódó publikációk: [J5]

A tervezési minták jelentőségén túl azok azonosítása és detektálása a projektekben éppúgy fontos, hogy a forráskódot és komponensek közötti kapcsolatokat jobban megértsük. Továbbá, értékes statisztikákat készíthetünk nagy mennyiségű projekt kiértékelésével. Ah-

hoz, hogy egy többcímkes osztályozó neurális hálóval magas pontosságot érjünk el, projektek ezreire lenne szükségünk. A manuális címkézés időigényes és hibaérzékeny, viszont az automatizált megoldások segíthetnek ezen. Napjainkban a prompt engineering még fel nem tárt lehetőségeket tartogat, amellyel a generatív mesterséges intelligencia (MI) használata kiterjeszhető. Az architektúrális tervezési mintákról lévő ismereteink alapján megterveztünk egy szabályhalmazt, amely segít egyértelműen elkülöníteni az MVW tervezési minta variánsait Angular projektekben. A GPT-3.5 és GPT-4 modellek már elég fejlettek ahhoz, hogy ne csak a kéréseinket értsék meg, hanem a válaszadáskor saját tudásukat is alkalmazzák. Szabályhalmazunk 30 szabályt tartalmaz, amelyek egymástól függetlenek és segítenek tisztán azonosítani az MVW tervezési minta variánsait. Megfigyeltük, hogy a GPT-4 sokkal jobban teljesít, és el is értünk egy 90%-os pontosságot a többcímkes osztályozási feladatunkban (lásd [7.](#) ábra). Az eredmények alapján felvázoltuk, hogy a különböző tervezési minták hogyan kerülnek alkalmazásra a gyakorlatban és implementációs javaslatokat adtunk meg Angular-ban.



7. ábra. A GPT modellek pontossága az MVW tervezési minták azonosításában

5. Kontribúciók

Az **első téziscsoportban** bemutattam különböző adatminőség és adatkonzisztencia elemző módszereket elosztott telemedicina rendszerekre, és kidolgoztam egy taxonómiát, amely az elosztott rendszerek kompromisszum problémájára kínál megoldást. A munka részletes kifejtése a 3. fejezetben található.

I / 1. Feltérképeztem a telemedicina rendszerek adatútjának komplexitását, és bemutattam, hogy a késleltetéssel az adatminőség és a konzisztencia csökkenhet. Javasoltam

a k -romlási paraméteret, amelyekkel a gyorsítótárak konfigurálhatóak elosztott telemedicina rendszerek esetében. Formális leírást készítettem egy ilyen rendszerről, továbbá a CAP és PACELC tételek mentén kidolgoztam egy taxonómiát az elosztott telemedicina rendszerek kompromisszum problémájára. A rendszermodellen szimulációkat végeztem, és kiértékeltem a rendszer működését különböző k -romlási paraméterek és késleltetési értékek mellett.

- I / 2. Bemutattam, hogy a modellezés és a szimuláció együttesével hatékonyan elemezhető az adatminőség telemedicina rendszerekben. Megmutattam, hogy a gráf reprezentáció, és a rendszereket leíró DAG struktúra kihasználásával az adatminőséget leíró állapottér komplexitása jelentősen csökkenthető.
- I / 3. Javasoltam egy közelítő módszert, amellyel egy rendszer folyamatainak a késleltetését leíró valószínűségi eloszlás függvény pontosan meghatározható. Lefejlesztettem egy keretrendszert, amellyel az adatminőséget leíró állapottér egy fókuszáltabb részét vizsgálhatjuk testreszabható körülmények között. Bemutattam a keretrendszer hatékonyságát valós algoritmusokat felhasználva. Demonstráltam, hogy a késleltetés következtében egy valós idejű telemedicina rendszernél szignifikáns adatminőség csökkenés következhet be.

Az **második téziscsoportban** a kontribúcióim a lefejlesztett SDK-hoz, a WebDAO tervezési minta bevezetéséhez és a fejlesztői produktivitás méréséhez kapcsolódnak. A 4. fejezet tárgyalja részletesen a munkákat.

- II / 1. Bemutattam, hogy a front-end és a back-end kapcsolata standardizálható a WebDAO tervezési mintával. Különösen a telemedicinában, az FHIR szabvány tökéletesen integrálható a WebDAO-val. Használati statisztikákat készítettem az FHIR szabvány használatáról a modern webfejlesztésben. Bemutattam, hogy a rétegelt architektúra hogyan változott a szervermentes paradigma megjelenésével, és demonstráltam, hogy az új rétegelt architektúra hogyan kerül alkalmazásra az Angular keretrendszerben.
- II / 2. Lefejlesztettem egy SDK-t telemedicina alkalmazások fejlesztéséhez, amely hibrid felhő támogatással bír és alkalmazza a WebDAO tervezési mintát. Javasoltam egy módszert, amellyel a fejlesztői produktivitás mérhető és számszerűsíthető. Valós fejlesztő bevonásával kiértékeléseket végeztem, és megmutattam, hogy a produktivitás jelentősen javítható a WebDAO tervezési minta használatával.
- II / 3. Bemutattam egy repository metrikákon alapuló produktivitás mérési módszert. Az eredmények alapján megállapítottam, hogy az Angular fejlesztésben, ha a WebDAO tervezési mintát alkalmazzák, a forráskódban történő változások száma jelentősen csökkenthető, és a hibajegyek megoldásához kevesebb idő szükséges.

A **harmadik téziscsoportban** bemutatott kontribúcióim a tervezési minták azonosítására és az elért eredményekre terjed ki NLP módszerekkel. Az 5. fejezet mutatja be teljeskörűen a munkát.

- III / 1. Kifejlesztettem két NLP módszert az Angular projektek és az Angular szolgáltatások osztályozására a WebDAO tervezési minta jelenlétének függvényében. Bemutattam, hogy a WebDAO-t implicit használják az Angular alkalmazásfejlesztésben és kimutattam, hogy számos projekt még csak szolgáltatásokat sem használ.
- III / 2. Bemutattam a Model-View-alapú architektúrális tervezési minták variánsait a modern webfejlesztésben. Kidolgoztam egy taxonómiát formális definícióval ellátva az MVW tervezési mintákra, illetve javaslatokat tettem a különböző tervezési minták megfelelő alkalmazására.
- III / 3. Bemutattam egy új prompt engineering módszert az MVW tervezési minták azonosítására Angular projektek esetében. Definiáltam egy szabályhalmazt, amely segít azonosítani az MVW tervezési mintákat Angular projektekben. Kiértékeltem mind a GPT-3.5 és a GPT-4 modelleket a szabály-alapú tervezési minta azonosítás témakörben.

A tézispontok és a publikációk kapcsolatát a [3.](#) táblázat foglalja össze.

3. táblázat. *Összefoglaló a tézispontok és a publikációk kapcsolatáról*

Publication	IF	SJR	Score	Thesis point								
				I/1	I/2	I/3	II/1	II/2	II/3	III/1	III/2	III/3
[J1]	-	Q3	0.75	•								
[J2]	-	Q3	1		•							
[J3]	2.9	Q2	0.75			•						
[J4]	2.6	Q2	0.75				•		•	•		
[J5]	2.9	Q2	0.75								•	•
[J6]	-	Q4	0.75				•	•				
[C1]	-	-	0.48					•				
[FR1]	-	-	-					•				
[FR2]	-	-	-	•								
[FR3]	-	-	-	•								
[FR4]	-	-	-					•				

A szerző publikációi a tézishez kapcsolódóan

Folyóirat publikációk

- [J1] **Z.R. Jánki** and V. Bilicki. Taxonomy for The Trade-off Problem in Distributed Telemedicine Systems. *Acta Cybernetica*, 25(2), pp. 285–306, 2021. SJR: Q3 **(0.75p)**
- [J2] **Z.R. Jánki**. A graph-based data quality analysis in distributed telemedicine systems. *Pollack Periodica*, 17(1), pp. 18–23, 2022. SJR: Q3 **(1p)**
- [J3] **Z.R. Jánki** and V. Bilicki. A Data Quality Measurement Framework Using Distribution-Based Modeling and Simulation in Real-Time Telemedicine Systems. *Applied Sciences*, 13(13), 7548, 2023. IF: 2.9 SJR: Q2 **(0.75p)**
- [J4] **Z.R. Jánki** and V. Bilicki. The Impact of the Web Data Access Object (WebDAO) Design Pattern on Productivity. *Computers*, 12(8), 149, 2023. IF: 2.6 SJR: Q2 **(0.75p)**
- [J5] **Z.R. Jánki** and V. Bilicki. Rule-based Architectural Design Pattern Recognition with GPT models. *Electronics*, 12(15), 3364, 2023. IF: 2.9 SJR: Q2 **(0.75p)**
- [J6] **Z.R. Jánki** and V. Bilicki. Standardized Telemedicine Software Development Kit with Hybrid Cloud Support. *Acta Cybernetica*, 2023. (accepted) SJR: Q4 **(0.75p)**

Konferencia kiadványok

- [C1] **Z.R. Jánki**, Z. Szabó, V. Bilicki and M. Fidrich. Authorization solution for full stack FHIR HAPI access. In *Proceedings of the IEEE 30th Neumann Colloquium (NC)*, pp. 121–124, 2017. **(0.48p)**

További kapcsolódó publikációk

- [FR1] **Z.R. Jánki** and V. Bilicki. Full-stack FHIR-based MBaaS with Server- and Client-side Caching Capable WebDAO. In *The 11th Conference of PhD Students in Computer Science*, pp. 192–196, 2018.

- [FR2] **Z.R. Jánki** and V. Bilicki. Crosslayer Cache for Telemedicine. In *The 12th Conference of PhD Students in Computer Science*, pp. 159–163, 2020.
- [FR3] **Z.R. Jánki** and V. Bilicki. The Trade-off Problem in Distributed Telemedicine Systems: A Case-study. In *The 16th Iványi Miklós International PhD & DLA Symposium*, 2020.
- [FR4] **Z.R. Jánki** and V. Bilicki. Domain Specific Semantic Data Model Integration. In *The 13th Conference of PhD Students in Computer Science*, pp. 196–200, 2022.

További publikációk

- [F1] Z. Szabó, V. Bilicki, Á. Berta and **Z.R. Jánki**. Smartphone-based data collection with stunner using crowdsourcing: lessons learnt while cleaning the data. In *Proceedings of International Academy, Research, and Industry Association*, 2017. **0.48p**
- [F2] Z. Szabó, V. Bilicki, Á. Berta and **Z.R. Jánki**. Smartphone-based data collection with Stunner, the reality of peer-to-peer connectivity and web real-time communications using crowdsourcing: Lessons learnt while cleaning the data. *International Journal On Advances in Software*, vol. 11, no. 1-2, pp. 120–130, 2018.
- [F3] A. Czégel, B. Lakos, **Z.R. Jánki**, A. Bánhalmi and V. Bilicki. Elektrokardiográfiai felvételek digitalizálása és kiértékelése okostelefonnal. *XXXII. Neumann Kollokvium*, 2019.
- [F4] **Z.R. Jánki** and V. Bilicki. Adatkonzisztencia és adatminőség elosztott telemedicina rendszerekben. *XXXIII. Neumann Kollokvium*, 2020.
- [F5] V. Bilicki, Z. Szabó and **Z.R. Jánki**. Számítógép hálózatok programozóknak. *University of Szeged*, 2020.

Total publication score: 5.71

Other References

- [1] Daniel Abadi. Consistency tradeoffs in modern distributed database system design: Cap is only part of the story. 45(2):37–42, 2012.

- [2] Peter Bailis, Shivaram Venkataraman, Michael J. Franklin, Joseph M. Hellerstein, and Ion Stoica. Probabilistically bounded staleness for practical partial quorums. *Proc. VLDB Endow.*, 5(8):776–787, apr 2012.
- [3] Eric Brewer. Cap twelve years later: How the "rules" have changed. *Computer*, 45(2):23–29, 2012.
- [4] Index - fhir v5.0.0. <http://hl7.org/fhir/>.
- [5] M. Fowler. *Patterns of Enterprise Application Architecture*. A Martin Fowler signature book. Addison-Wesley, 2003.
- [6] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series. Pearson Education, 1994.
- [7] Inclouded. <http://inclouded.hu/>.
- [8] Mayuram S Krishnan, Charlie H Kriebel, Sunder Kekre, and Tridas Mukhopadhyay. An empirical analysis of productivity and quality in software products. *Management science*, 46(6):745–759, 2000.
- [9] Leslie Lamport, John Matthews, Mark Tuttle, and Yuan Yu. Specifying and verifying systems with tla+. In *Proceedings of the 10th Workshop on ACM SIGOPS European Workshop, EW 10*, page 45–48, New York, NY, USA, 2002. Association for Computing Machinery.
- [10] Harlan D Mills. *Software productivity*. 1988.
- [11] Mark Richards. *Software architecture patterns*. O’Reilly Media, 2015.
- [12] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*. John Wiley & Sons, 2016.
- [13] Stefan Wagner and Melanie Ruhe. A systematic review of productivity factors in software development. *CoRR*, abs/1801.06475, 2018.
- [14] Yuan Yu, Panagiotis Manolios, and Leslie Lamport. Model checking tla+ specifications. In *Advanced Research Working Conference on Correct Hardware Design and Verification Methods*, pages 54–66. Springer, 1999.
- [15] Cheng Zhang and David Budgen. What do we know about the effectiveness of software design patterns? *IEEE Transactions on Software Engineering*, 38(5):1213–1231, 2012.

6. Summary

The theses presents the novelties across three thesis groups. In the first thesis group, we introduce the distributed telemedicine systems and our modeling approach to simulate their operation in the focus of the data. The second thesis group presents an SDK and its impact on the development productivity. In the third thesis group, we introduce the design patterns applied in the modern Web development and various NLP techniques to identify the patterns.

For a system to operate as expected, we must understand the specific use-case. Using a formal system model and simulation, we discovered that the allowable latency determined by the use-case significantly dictates the system capability trade-offs we must make. In addition, the simulation results showed that the latency can materially reduce the data quality. We elaborated a graph-based clustering solution to analyze the data quality. Furthermore, we implemented a focused simulation environment based on the emulation results and the approximation techniques.

Throughout our telemedicine developments, we have gained substantial experience in modern web development and its associated areas. The significance of using software design patterns has never been in doubt. However, new technologies and frameworks might implement modifications to well-established concepts. By introducing and applying the WebDAO design pattern, we achieved noticeable improvements in productivity, which we quantified in our research. Building on initial successes, we explored the pattern's global presence and its impact on productivity.

Finally, we analyzed how the well-known Model-View-based design patterns changed in the modern Web development. We found that we can achieve promising results in design pattern detection using NLP methods.

All three domains still contain unanswered questions and present numerous new research avenues.

Nyilatkozat

Jánki Zoltán Richárd *Telemedicine Engineering: Modeling and Analyzing Data Quality in Telemedicine Systems and the Impact of Novel Design Patterns on Productivity in Modern Web Application Development* című PhD disszertációjában a következő eredményekben Jánki Zoltán Richárd hozzájárulása volt a meghatározó:

I/1. tézispont:

1. Adatkonzisztencia és adatminőség mérési módszerek definiálása. Elosztott telemedicina rendszer formális specifikációja. Szimulációs eredmények az adatkonzisztencia változására a gyorsítótár k -romlási paraméterének változtatása mentén. Elosztott telemedicina rendszerek taxonómiájának megalkotása.
 - Z. R. Jánki and V. Bilicki, Taxonomy for Trade-off Problem in Distributed Telemedicine Systems, *Acta Cybernetica*, vol. 25, no. 2, pp. 285–306, Aug. 2021. SJR: Q3
2. Adatkonzisztencia vizsgálatának formális leírása elosztott telemedicina rendszereknél. Elosztott telemedicina rendszerekhez paraméterezhető gyorsítótárazási modell implementálása. Szimulációs eredmények az adatkonzisztencia változására a gyorsítótár k -romlási paraméterének változtatása mentén.
 - Z.R. Jánki and V. Bilicki. Crosslayer Cache for Telemedicine. In *The 12th Conference of PhD Students in Computer Science*, pp. 159—163, 2020.
3. A CAP- és PACELC-tételben ismertetett kompromisszum problémájának (trade-off problem) demonstrálása egy telemedicina használati esettel.
 - Z. R. Jánki and V. Bilicki, The Trade-off Problem in Distributed Telemedicine Systems: A Case-study, In *The 16th Iványi Miklós International PhD & DLA Symposium*, 2020.

I/2. tézispont:

1. Formális telemedicina rendszerspecifikáció készítése. Gráf-alapú adatminőség mérése elosztott telemedicina rendszereknél. DAG struktúra megállapítása és a gráf gyengén összefüggő komponenseinek klaszterezése az adatminőségi értékek mentén.
 - Z. R. Jánki, A graph-based data quality analysis in distributed telemedicine systems, *Pollack Periodica*, vol. 17, no. 1, pp. 18–23, 2022. SJR: Q3

I/3. tézispont:

1. A kiválasztott használati eset rendszerének formális leírása. A valós idejű telemedicina rendszereknél potenciálisan felmerülő adatminőség csökkenés problémájának felvázolása. Emulációs környezet kiépítése a rendszer folyamatainak futtatásához. Emulációs eredmények mentén közelítő módszerek alkalmazásával egy fókuszált állapotter kialakítása. Szimulációs keretrendszer megvalósítása az állapotter vizsgálatához és az eredmények kiértékelése.
 - Z. R. Jánki and V. Bilicki, A Data Quality Measurement Framework Using Distribution-Based Modeling and Simulation in Real-Time Telemedicine Systems, *Applied Sciences*, vol. 13, no. 13, 7548, 2023. IF: 2.9 SJR: Q2

II/1. tézispon:

1. WebDAO tervezési minta bemutatása a modern Web fejlesztésben.
 - Z. R. Jánki and V. Bilicki, The Impact of the Web Data Access Object (WebDAO) Design Pattern on Productivity, Computers, vol. 12, no. 8, 149, 2023. IF: 2.6 SJR: Q2
2. Statisztikák készítése az FHIR szabvány használatáról nyílt forrású projektekben. Az Included SDK képességeinek bemutatása: hibrid felhő támogatás, polyglot perzisztencia megvalósítása, szabványos FHIR interfészek implementálása.
 - Z. R. Jánki and V. Bilicki, Standardized Telemedicine Software Development Kit with Hybrid Cloud Support, Acta Cybernetica, 2023. (elfogadva) SJR: Q4

II/2. tézispon:

1. A WebDAO tervezési minta implementálása az Included SDK formájában. A produktivitás javulásának demonstrációja valós fejlesztők bevonásával és az eredmények kiértékelése.
 - Z. R. Jánki and V. Bilicki, Standardized Telemedicine Software Development Kit with Hybrid Cloud Support, Acta Cybernetica, 2023. (elfogadva) SJR: Q4
2. Mobile Backend as a Service (MBaaS) megvalósítás egy SDK formájában szabványos FHIR interfésszel.
 - Z. R. Jánki, Z. Szabó, V. Bilicki and M. Fidrich, Authorization solution for full stack FHIR HAPI access, 2017 IEEE 30th Neumann Colloquium (NC), Budapest, Hungary, 2017, pp. 121–124, doi: 10.1109/NC.2017.8263266.
3. Gyorsítótárazást támogató MBaaS megvalósítás a WebDAO tervezési minta alkalmazásával.
 - Z. R. Jánki, and V. Bilicki, Full-stack FHIR-based MBaaS with Server- and Client-side Caching Capable WebDAO, The 11th Conference of PhD Students in Computer Science (CSCS), Szeged, Hungary, 2018, pp. 179–183.
4. FHIR szabvány bővíthetőségének támogatása NLP megoldásokkal.
 - Z. R. Jánki and V. Bilicki, Domain Specific Semantic Data Model Integration, In The 13th Conference of PhD Students in Computer Science, 2022. pp. 196–200.

II/3. tézispon:

1. A WebDAO tervezési minta alkalmazása a modern web-fejlesztésben. A WebDAO produktivitásra gyakorolt hatásaival kapcsolatos eredmények az Angular alkalmazásfejlesztésben.
 - Z. R. Jánki and V. Bilicki, The Impact of the Web Data Access Object (WebDAO) Design Pattern on Productivity, Computers, vol. 12, no. 8, 149, 2023. IF: 2.6 SJR: Q2

III/1. tézispont:

1. A WebDAO tervezési minta osztályozására implementált Random Forest osztályozó. A WebDAO tervezési minta osztályozására implementált BERT modell. A WebDAO Angular keretrendszerben történő alkalmazásával kapcsolatos statisztikai eredmények.

- Z. R. Jánki and V. Bilicki, The Impact of the Web Data Access Object (WebDAO) Design Pattern on Productivity, Computers, vol. 12, no. 8, 149, 2023. IF: 2.6 SJR: Q2

III/2. tézispont:

1. Model-View-Whatever (MVW) tervezési mintacsalád taxonómiájának megalkotása, a minták formális definiálása. MVW tervezési minták használatára javaslatok és implementációs minták kidolgozása Angular keretrendszerben.

- Z. R. Jánki and V. Bilicki, Rule-based Architectural Design Pattern Recognition with GPT models, Electronics, vol. 12, no. 15, 3364, 2023. IF: 2.9 SJR: Q2

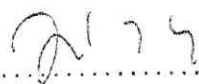
III/3. tézispont:

1. Angular keretrendszerre épített szabályrendszer kidolgozása, amely az MVW tervezési mintákat egyértelműen definiálja. GPT-modellek kiértékelése és összehasonlítása az MVW tervezési minták azonosítására Angular projektekben.

- Z. R. Jánki and V. Bilicki, Rule-based Architectural Design Pattern Recognition with GPT models, Electronics, vol. 12, no. 15, 3364, 2023. IF: 2.9 SJR: Q2

Ezek az eredmények Jánki Zoltán Richárd PhD disszertációján kívül más tudományos fokozat megszerzésére nem használhatók fel.

Szeged, 2023. augusztus 8.



Jánki Zoltán Richárd
jelölt



Dr. Bilicki Vilmos
témavezető

Az Informatika Doktori Iskola vezetője kijelenti, hogy jelen nyilatkozatot minden társszerzőhöz eljuttatta, és azzal szemben egyetlen társszerző sem emelt kifogást.

Szeged, 2023. 08. 21.





Dr. Jelasity Márk
Informatika Doktori Iskola vezetője

