

Informatikai rendszertervezés

HF3: Viselkedés modellezés

FMT csapat

Csapattagok

Formanek András	XXXXXX	xx@
Tegzes Tamás	YYYYYY	yy@
Manninger Miklós	ZZZZZZ	zz@

2017. Október 23.

Tartalomjegyzék

Tartalomjegyzék	1
1. Feladat ismertetése	2
2. Feladatok végrehajtása	3
2.1 Activity diagram készítése	3
2.1.1 Új route igénylése és készítése	3
2.2 Állapotgép diagramok készítése	4
2.2.1 Smart Train Controller állapotgépe	4
2.2.2 Schedule Manager állapotgépe	5
2.3 Interaction diagramok	6
2.3.1 Más diagramok részének kifejtése	6
2.3.2 A Scheduling Manager és a Smart Train Controller közti kommunikációt leíró diagram	9
2.4 Előző diagramok validálása Cameo Simulation Toolkit segítségével	10
3. Megjegyzések	11
4. Munkanapló	11

1. Feladat ismertetése

A feladat szövegét sok helyen nem éreztük egyértelműnek, ezért leírjuk, hogy hogyan értelmeztük az egyes mondatokat, a feladatok részleteit:

Az ügyfelek kérelmezhetik, hogy az árujukat a már megtervezett út helyett két vagy több állomáson keresztül egy általuk meghatározott más útvonalon szállítsuk. Az ilyen kérelmeket részben az operátor felelőssége elbírálni, részben a rendszer automatikusan állapítja meg teljesíthetőségüket. Az operátornak van lehetősége elutasítani a kérelmet.

A Smart Transportation System egyik kulcsfunkciója, hogy dinamikusan képes egyrészt elosztani a vonatokat a kiindulási- és célpontjuk között vezető útvonalakon (egyenletes kihasználás elérése végett), másrészt mentesítő járatot indítani, ha egy Service Line-on a kapacitásánál nagyobb mennyiségű árut kell szállítani.

A mentesítő járat indításának az előfeltétele, hogy legyen legalább egy szabad vonat (mozdony és kellő mennyiségű vagon), és hogy az útvonal menti állomásokon legyen olyan időszáv, amikor az új vonat ott megállhat, és le- illetve felszállhatnak az utasok.

A Smart Train Controller továbbá felelős a vonatok állomás közötti útvonalának tervezéséért, és a menetrend betartásáért.

Minden ügyféltől érkezett kérelem beteljesítendő, amint annak az összes előfeltétele adott, melyek egyike az operátor beleegyezése.

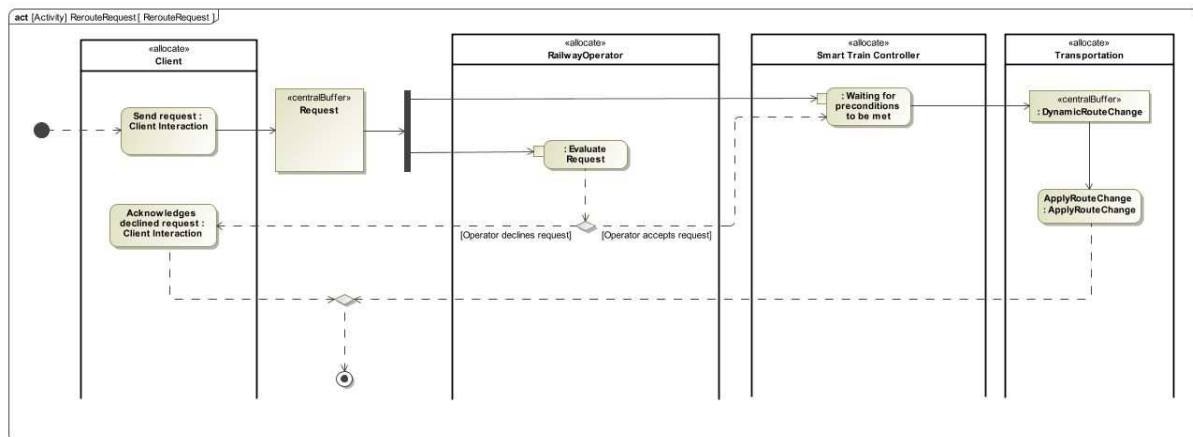
Az eheti feladat ismertetése:

- a. Feladatunk activity diagramokon bemutatni az ügyfél útvonalváltoztatási kérésének feldolgozását. Beleértve a kérés indítását, a kérés feldolgozását, és végrehajtását.
- b. Ezt követően készítenünk kell állapotgép diagramokat, melyeken a Smart Train Controller és a Schedule Manager viselkedését fel.
- c. Feladatunk továbbá Interaction diagramok készítése, mely a Smart Train Controller és a Schedule Manager közti interakciókat mutatja.
- d. Végül validálnunk kell a modelleinket az a), b) és c) feladatból a Cameo Simulation Toolkit segítségével.

2. Feladatok végrehajtása

2.1 Activity diagram készítése

2.1.1 Új route igénylése és készítése



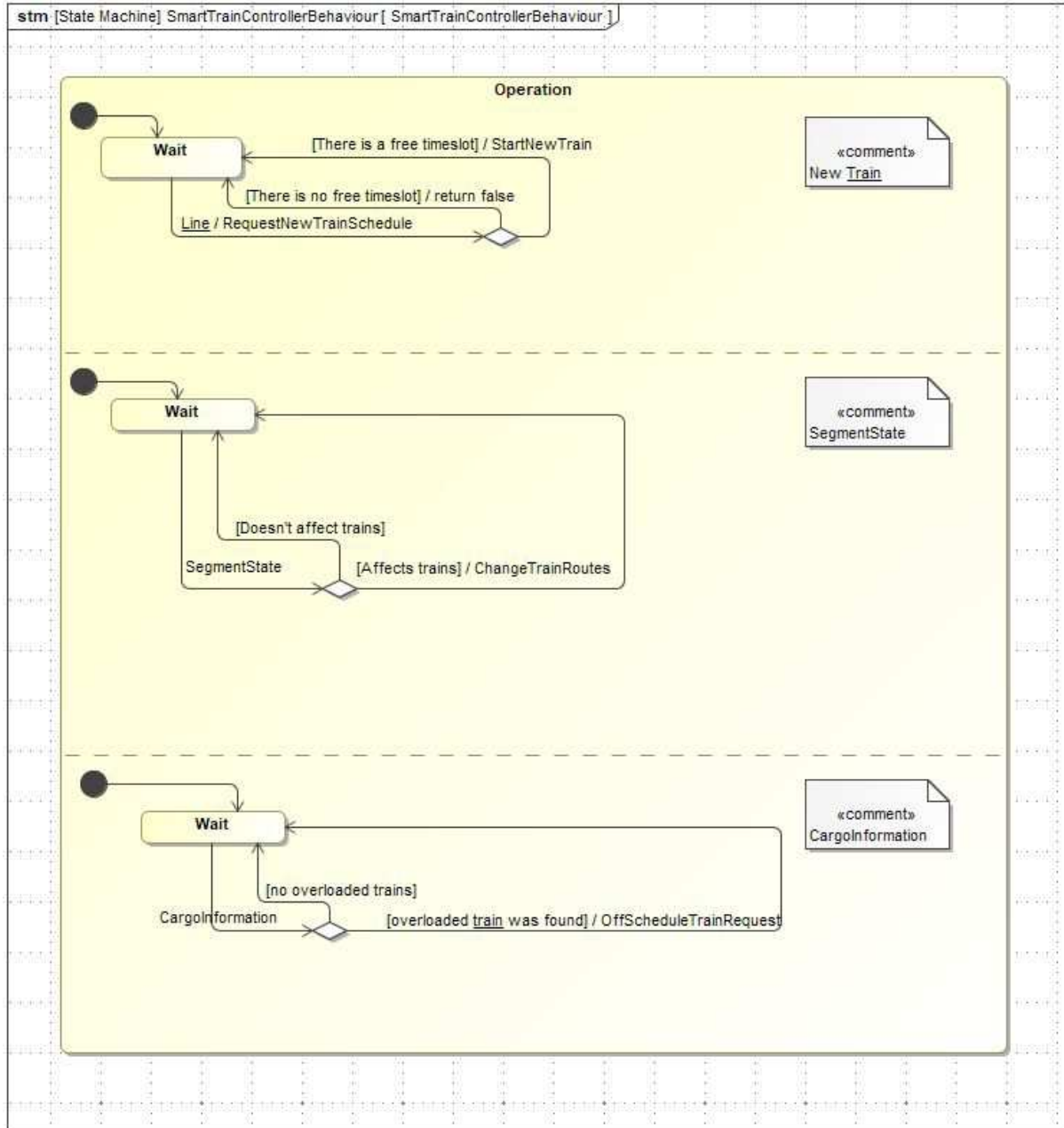
1. ábra

Az 1. ábra mutatja az ügyfél útvonalváltóztatási kérésének feldolgozását. Az ügyfél kérése eljut a RailwayOperatorhoz és a SmartTrainControllerhez egyszerre, előbbi dönt arról, hogy a változtatást elfogadjuk-e, utóbbi pedig amint minden előfeltétel adott (beleértve az operátor jóváhagyását), végrehajtja a változtatást. Ha az operátor nem hagyja jóvá a változtatást, a rendszer értesíti az ügyfelet, és nem hajtja végre a módosítást.

Az ügyféllel való kommunikációt a 7. ábrán ábrázoltuk.

2.2 Állapotgép diagramok készítése

2.2.1 Smart Train Controller állapotgépe

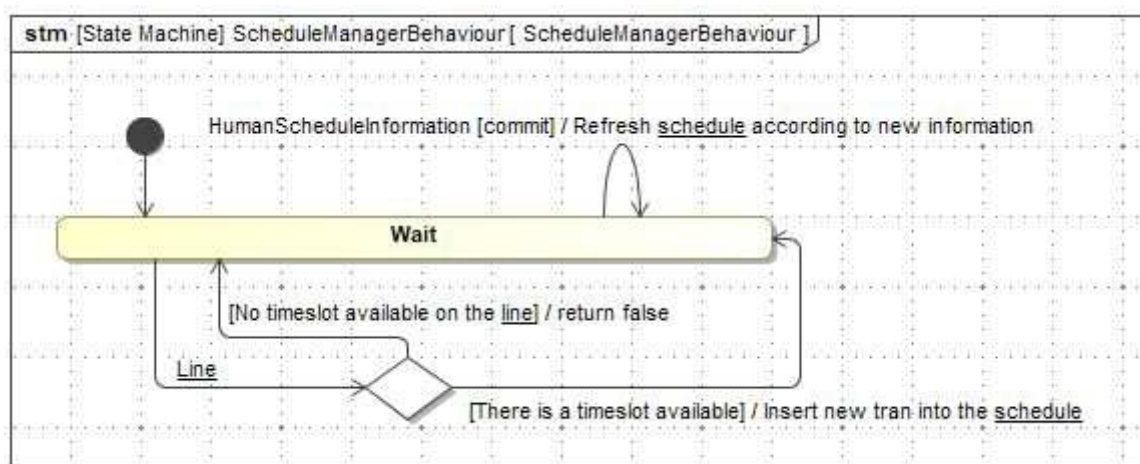


2. ábra

A 2. ábra mutatja a Smart Train Controller állapotgépét. Ez a blokk mindig a beérkező jelekre vár, nincs más állapota. A diagram három párhuzamos részből áll:

1. NewTrain
A Line jel hatására, ha van időkeret az új vonatnak, akkor indít egy új vonatot (4.ábra), különben visszajelezi, hogy nem sikerült indítani.
2. SegmentState
A SegmentState jel hatására megváltoztatja a vonatok útvonalát (5.ábra), amennyiben a vonatok érintettek. Ha egy vonat sem érintett, akkor nem csinál semmit.
3. CargoInformation
A CargoInformation jel alapján, ha talált túlterhelt vonatot, akkor mentesítő-járatot kér (6.ábra). Ha nem talált ilyen vonatot, akkor nem csinál semmit.

2.2.2 Schedule Manager állapotgépe



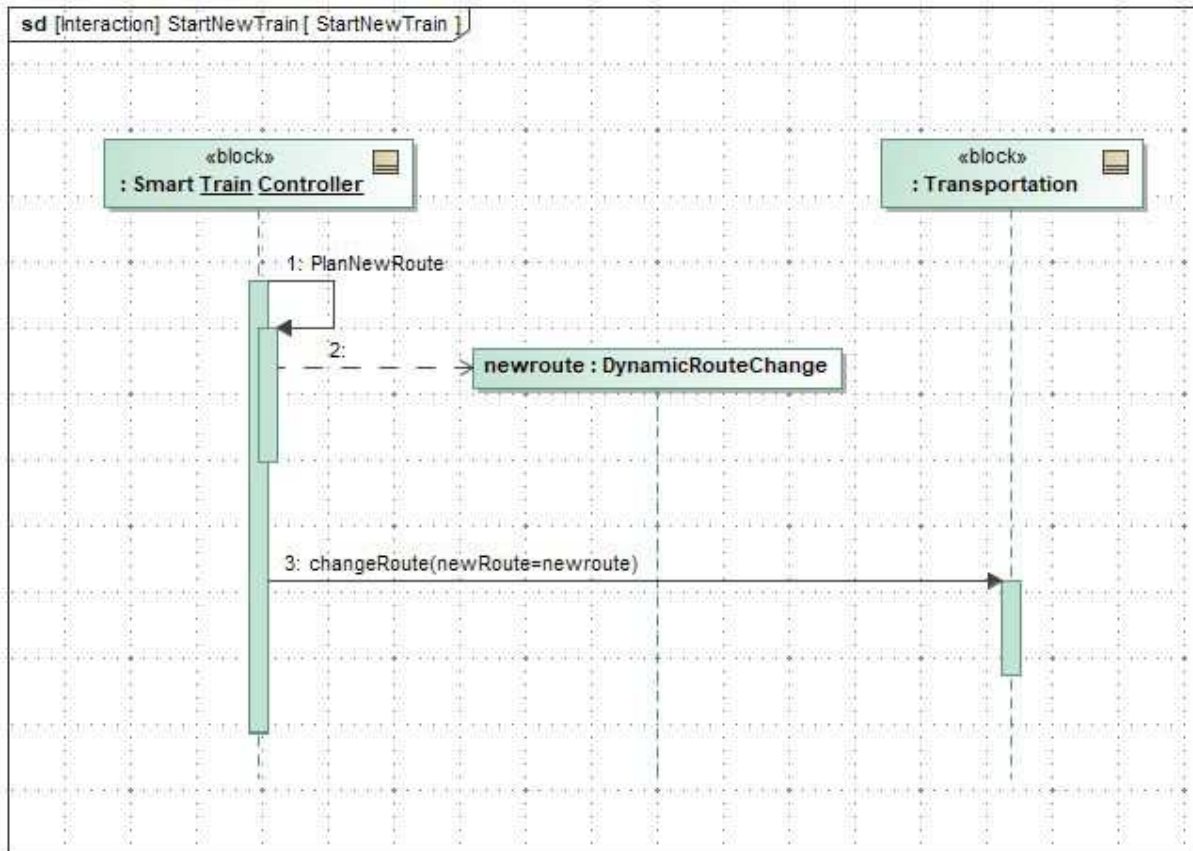
3. ábra

A 3. ábra mutatja a Schedule Manager állapotgépét. Ez a blokk mindig a beérkező jelekre vár, nincs más állapota. Ha HumanScheduleInformation érkezik, ami véglegesíti az addig beérkezett információkat, a mindenkor érvényes menetrend frissül. Ha Line jel érkezik, annak hatására a kapott vonalra új vonatot próbál a menetrendbe venni. Ha van szabad időszáv minden állomáson, akkor beleveszi az új vonatot a menetrendbe, különben nem csinál semmit.

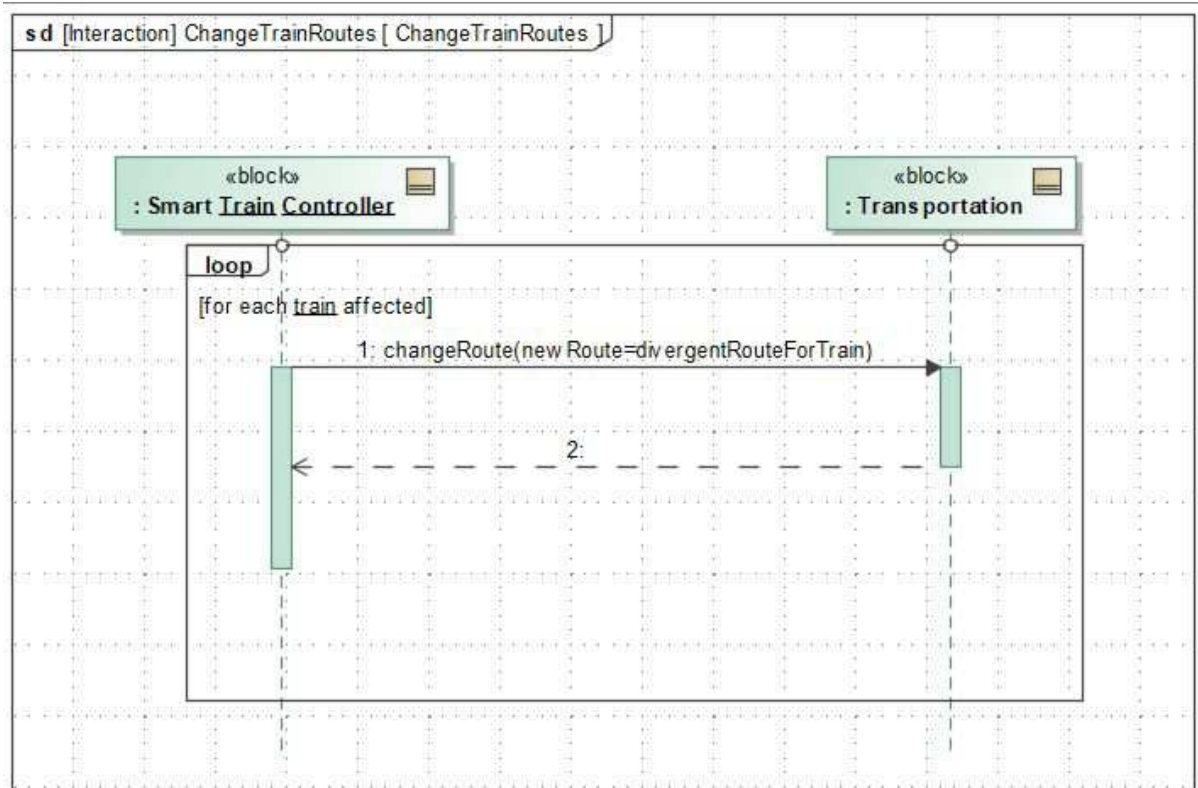
2.3 Interaction diagramok

Ebben a fejezetben találhatóak az interaction diagramok. A legtöbb diagram egy másik fejezetben egy-egy hivatkozásként megtalálható.

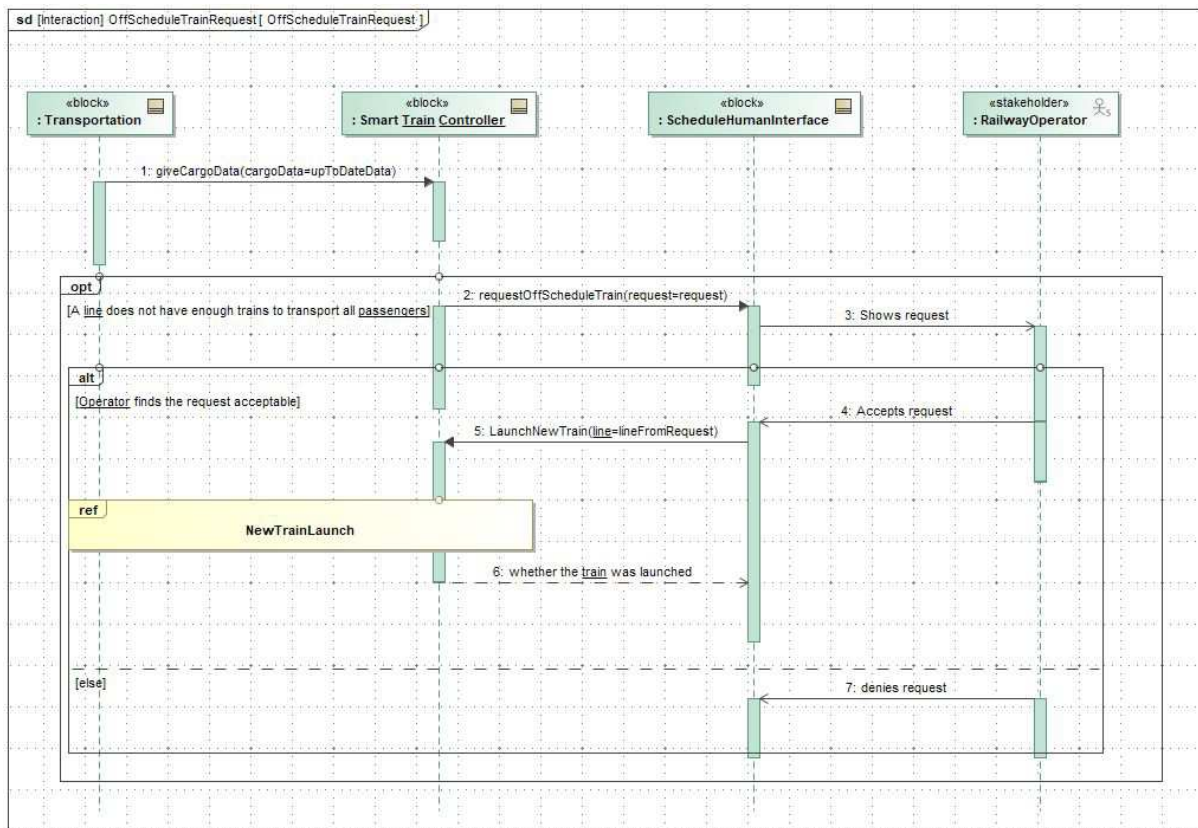
2.3.1 Más diagramok részének kifejtése



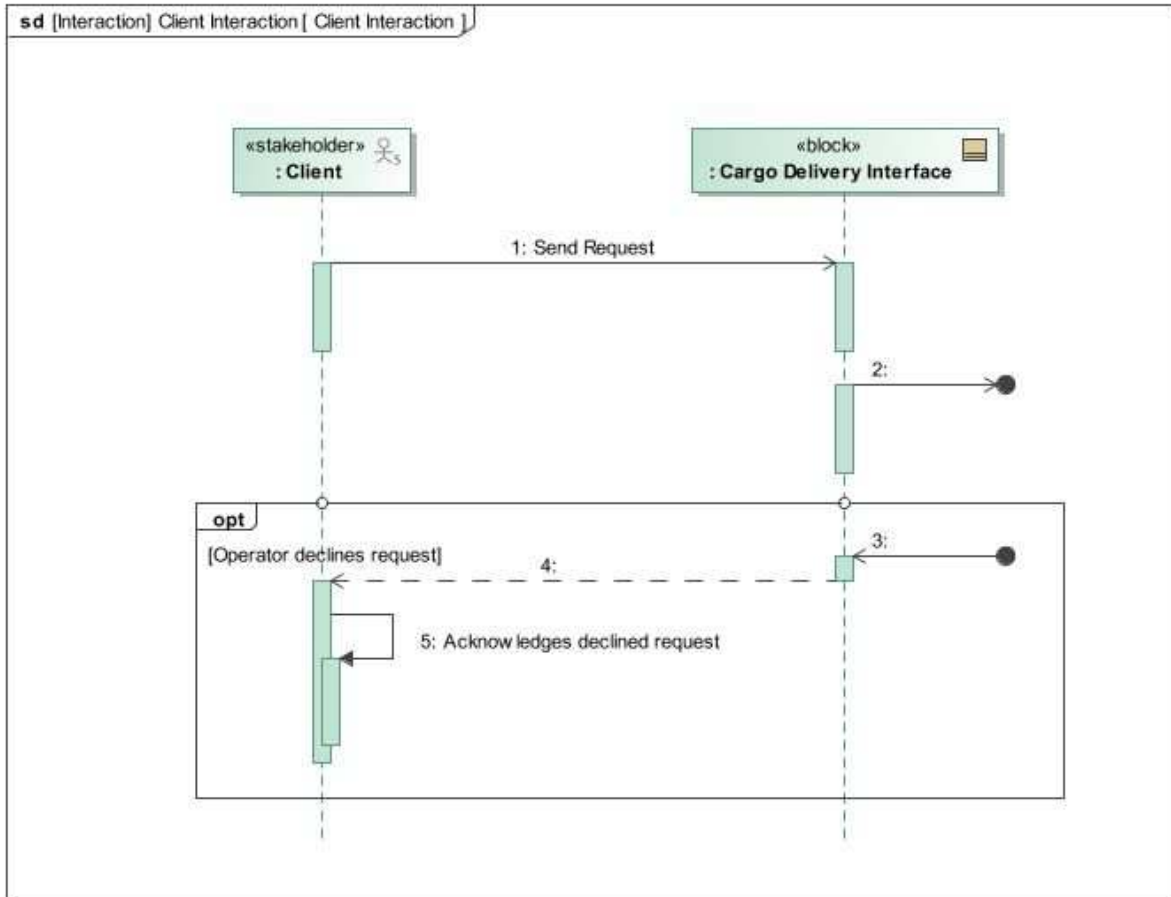
4. ábra



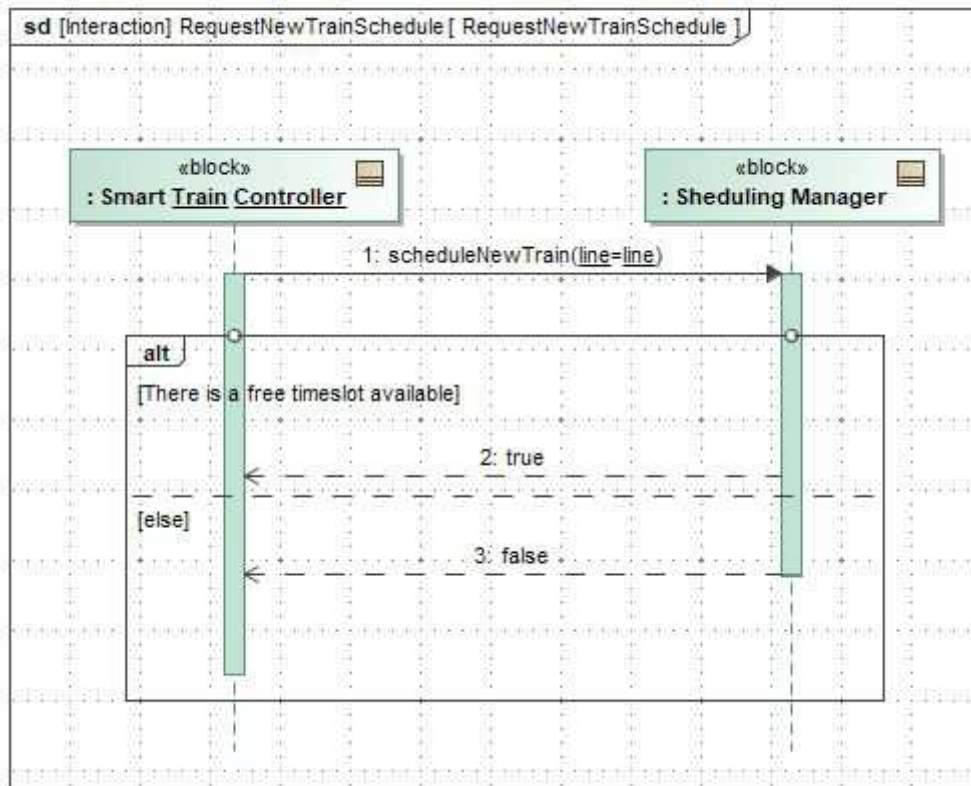
5. ábra



6. ábra

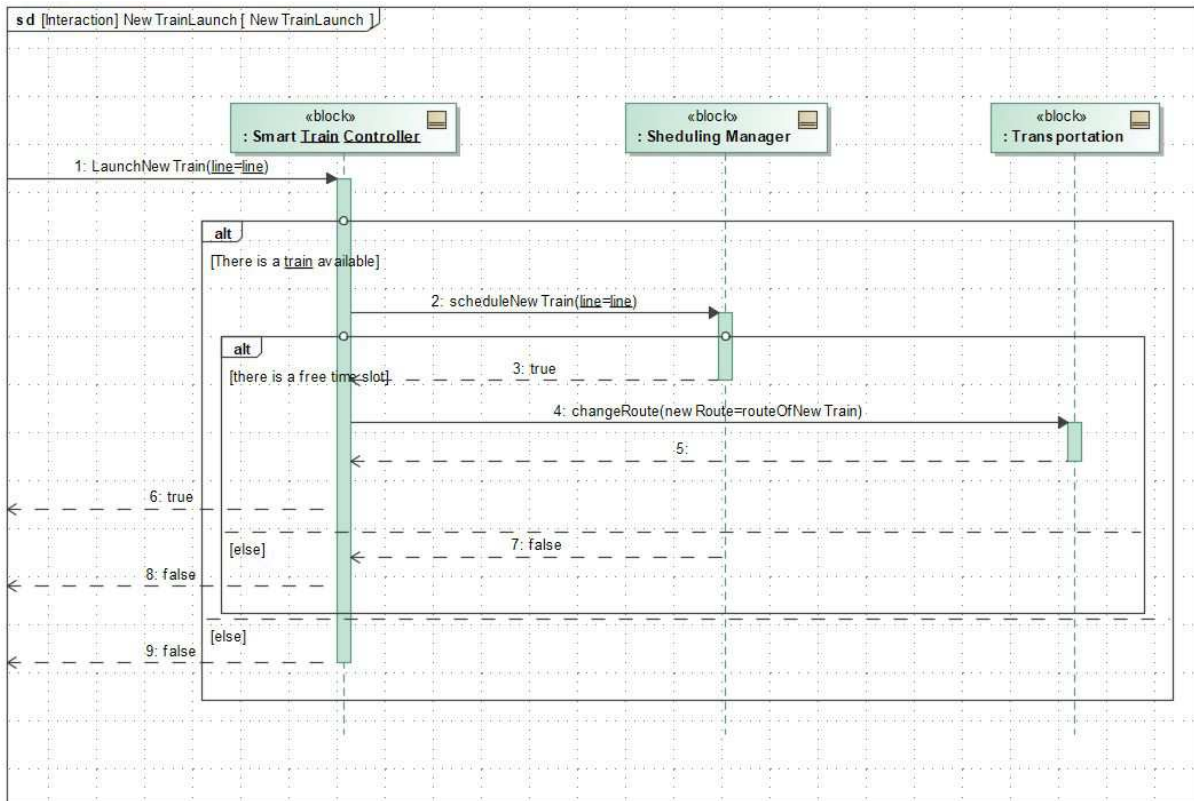


7. ábra



8. ábra

2.3.2 A Scheduling Manager és a Smart Train Controller közti kommunikációt leíró diagram



9. ábra

2.4 Előző diagramok validálása Cameo Simulation Toolkit segítségével

1. Ábra

A szimulátor hibát adott, mert az általunk használt Central Buffer típusú object node nem támogatott. Mi viszont nem tudtunk a diagramra olyan object node-ot tenni, ami nem rendelkezik ezzel a sztereotípiával. Levettük ezért ezeket a node-okat a szimuláció idejére, ekkor a fork-on akadt el a futás, számunkra érthetetlen okból, ha ezt is eltávolítottuk, és helyette külön object flowt indítottunk, akkor már működött.

A ClientInteraction szekvenciadiagramot úgy döntöttünk, hogy egybe ábrázoljuk, holott két külön része fut le az activity diagram futtatása során. A szimulációban értelem szerűen kétszer lefut az egész, mi úgy gondoljuk, hogy ez mégsem okoz kétértelműséget.

2. Ábra

A szimulátor segítségével sikerült kijavítanunk a diagramot. A hiba az volt, hogy a diagramról törölt, de a fában még létező objektumok megkavarták a szimulátort. Az Opaque behavior-öket nem tudta végrehajtani.

3. Ábra

A szimulátor hibát ad, mert mi Opaque behavior-rel jelöltük azokat az viselkedéseket, amelyeket nem kívántunk bővebben definiálni, ezeket nem tudta kiértékelni. Annyit megtudtunk, hogy az általunk gondolt pillanatban próbálja meghívni a megfelelő behavior-öket.

4. Ábra

Kapunk egy warningot, miszerint a **Lifeline objektum nem található**. Ennek ellenére a szimuláció lefut. Mi ezeket az objektumokat singleton-ként képzeljük el, és mindig egy port által megvalósított interfészen keresztül kommunikálunk egymással.

Az alternatív combined fragment mindkét ága lefutott egymás után, felhetőleg azért, mert a magas szintű feltételt nem tudta kiértékelni a szimulátor.

5. Ábra

Itt is kaptunk Lifeline object can't be found. A szimuláció a loop magasszintű feltételét nem tudja kiértékelni, így végtelenül fut.

6. Ábra

A szimuláció lefutott, de a referált diagramrészbe nem lépett be a szimuláció, számunkra érthetetlen okokból.

7. Ábra

A diagram hiba nélkül lefutott.

8. Ábra

A diagramon csak Lifeline object can't be found hibát tapasztaltunk, az alternatív fragmentnek itt is lefutott mindkét ága egymás után.

9. Ábra

Az eddigiekkel egyezőket tapasztaltuk, Lifeline object can't be found warningok, és az alternatív fragmentek mindkét ága lefut.

3. Megjegyzések

Hasznosnak találnánk olyan alkalmak tartását, amikor a MagicDraw segítségével készítünk el valamilyen mintamodellt, ugyanis a viselkedésmodellezési eszközök már elég bonyolultak, sokszor részletesebb specifikációt várnak el, mint amit nekünk egyébként feladatunk lenne megadni. Pl. Opaque Behavior-t nem engedett egyáltalán használni, így Activity és StateChart diagramokon nehéz magas szintű cselekvést vagy feltételt leírni. A gyakorlatokon viszont csak magas szintű dolgokat használtunk, ezért gondoltuk, hogy a háziban is megfelelőek ezek.

4. Munkanapló

Kezdet	Időtartam	Résztevők	Leírás
2017-10-14	30 perc	Manninger	Dokumentáció szkeletonjának elkészítése
2017-10-14	2 óra 30 perc	Tegzes, Formanek, Manninger	Skype beszélgetés: Közös feladatmegoldás b) feladat
2017-10-19	2 óra 30 perc	Tegzes, Formanek, Manninger	Közös konzultáció: A feladat szövegének értelmezése, újragondolása, feladatok kiadása
2017-10-21	1 óra	Tegzes	CreateNewRoute és hivatkozott diagramok
2017-10-21	2 óra	Tegzes, Formanek, Manninger	Skype konzultáció: iteratív inkrementális átnézés az alapoktól, néhány változtatás,

2017-10-21	15 perc	Tegzes, Formanek, Manninger	Szünet
2017-10-21	3 óra	Tegzes, Formanek, Manninger	Skype konzultáció folytatása, végül feladatok kiosztása
2017-10-23	1 óra 30 perc	Manninger	Dokumentáció készítése, kisebb hibák javítása
2017-10-23	1 óra	Tegzes	Dokumentáció, OffScheduleTrainRequest
2017-10-23	1 óra 40 perc	Tegzes, Manninger	Közös dokumentáció készítés, régebbi dokumentációk konzisztensé tétele
2017-10-23	1 óra	Tegzes, Formanek	Diagramok validálása
2017-10-23	1 óra 45 perc	Tegzes, Manninger, Formanek	Második házi feladat szorgalmi készítése: jólfomáltsági
2017-10-23	30 perc	Tegzes, Manninger, Formanek	Végleges beadáshoz a három doksi ellenőrzése.