

4. Analízis modell kidolgozása 2

4.1 Objektum katalógus

4.1.1 Bank

A játék startpozíciója, innen indul a rabló.

4.1.2 Building

A városban található épületeket megvalósító objektum.

4.1.3 Car

A városban közlekedő autókat megvalósító objektum.

4.1.4 City

A város, magában foglalja az öt felépítő elemeket és felelős a mozgásukért.

4.1.5 Hideout

A menedék épülete, a cél pozíciót testesíti meg.

4.1.6 ITraffic

A közlekedési szabályok szolgáltatásaiért felelős interface.

4.1.7 Police

A rendőrt megtestesítő objektum. Üldözik a rablót.

4.1.8 RoadBlock

Az utak építőeleme, nyilvántartja a rajta közlekedő autókat és közlekedési táblákat, lámpákat.

4.1.9 Robber

A rabló, őt irányíthatjuk a játék során.

4.1.10 TrafficSign

A jelzőlámpa, szabályozza a forgalmat a városban.

4.1.11 TrafficTable

A jelzőtáblák, szabályozza a forgalmat a városban.

4.2 Osztályok leírása

4.2.1 Bank

- **Felelősség**

Jelezni a játékosnak a játék végét, ha elérte.

- **Ősosztályok**

Building

- **Interfészek**

nincs

- **Attribútumok**

- **int Role** : megadja az épület szerepét a játékban

- **Metódusok**

- **int getRole()** : visszaadja az osztály szerepét jelentő belső változó értékét

4.2.2 Building

- **Felelősség**

A városban lévő épületek, felelősségük a szerepkörük nyilvántartása.

- **Ősosztályok**

nincs

- **Interfészek**

nincs

- **Attribútumok**

- **int role**: szerepkör jelölése

- **Metódusok**

- **int getRole()**: szerepkör lekérdezése

4.2.3 Car

- **Felelősség**

A városban közlekedő autók, felelősségük a szabályok betartása és az ütközések elkerülése, sebességük nyilvántartása.

- **Össosztályok**

nincs

- **Interfészek**

nincs

- **Attribútumok**

- **int speed** : a kocs sebessége, hány „steppenként” lép az autó
- **int timetomove** : várakozási idő lámpák és táblák esetén

- **Metódusok**

- **int step()** : a mozgást valósítja meg
- **void move()** : az ütközésselkerülést segítő metódus
- **void pass(Car)** : az előzést végző függvény
- **void setSpeed(int)** : beállítja az autó sebességét
- **int getSpeed()** : visszaadja az autó sebességét
- **void destroy()** : törli az autót

4.2.4 City

- **Felelősség**

A várost reprezentáló osztály, felelőssége a városba be és kilépések szabályozása, a városban lévő objektumok nyilvántartása, mozgatása.

- **Össosztályok**

nincs

- **Interfészek**

nincs

- **Attribútumok**

- **LinkedList<Car> car** : a városban közlekedő autók
- **LinkedList<RoadBlock> road** : a város útszerkezete
- **LinkedList<ITraffic> traffic** : a városban lévő közlekedési szabályok
- **LinkedList<Building> building** : a városban található épületek

- **Metódusok**
 - **void step()** : a város mozgatása

4.2.5 Hideout

- **Felelősség**
Jelezni a játékosnak a játék végét, ha elérte.
- **Ősosztályok**
Building
- **Interfészek**
nincs
- **Attribútumok**
 - **int Role** : az épület szerepköre
- **Metódusok**
 - **int getRole()** : visszaadja az épület szerepkörét

4.2.6 ITraffic

- **Felelősség**
A közlekedés szabályozása, interfacet nyújt a közlekedési szabályokat megvalósító osztályok számára.
- **Ősosztályok**
nincs
- **Interfészek**
nincs
- **Attribútumok**
nincs
- **Metódusok**
 - **int getState()** : megadja a jelzés állapotát
 - **void step()** : a váltásokat végző függvény

4.2.7 Police

- **Felelősség**
A rendőrt reprezentáló objektum. Felelőssége ha elkapta a rablót megszakítani a játékot.

- **Ősosztályok**
Car
- **Interfészek**
nincs
- **Attribútumok**
- **Metódusok**
 - **void pass(Robber)** : ha egy rabló próbálja megelőzni akkor azt elkapja

4.2.8 RoadBlock

- **Felelősség**
Az utat építi fel, felelőssége a rajta található autók , jelzőlámpák, -táblák és az őt körülvevő RoadBlockok nyilvántartása.
- **Ősosztályok**
nincs
- **Interfészek**
nincs
- **Attribútumok**
 - **Car& car** : a rajta található autóra mutató referencia
 - **ITraffic[] traffic** : a rajta található táblák, és lámpák
 - **Building building** : a rajta található épületek
 - **RoadBlock road** : a vele szomszédos útdarabok
- **Metódusok**
 - **void setNeighbour(Neighbour)** : beállítja a szomszédos útdarabokat
 - **void setCar(Car)** : beállítja a rajta lévő autót
 - **void setBuilding(Building)** : beállítja a hozzá tartozó épület
 - **void setTraffic(ITraffic)** : beállítja a hozzá tartozó lámpákat, táblákat
 - **RoadBlock[] getNeighbour()** : visszaadja a szomszédos útdarabokat
 - **Car getCar()** : visszaadja a rajta lévő autót
 - **Building[] getBuilding()** : visszaadja a hozzá tartozó épület
 - **ITraffic& getTraffic()** : visszaadja a hozzá tartozó lámpákat, táblákat

4.2.9 Robber

- **Felelősség**
A rabló, őt irányíthatjuk a játék során, felelőssége a rendőr detektálása.
- **Ősosztályok**
Car

- **Interfészek**
nincs
- **Attribútumok**
- **Metódusok**
 - **void pass(Car)** : az előtte haladó autó megelőzése
 - **void step()** : a mozgása eltér a többi járművétől, hiszen nem vonatkoznak rá szabályok

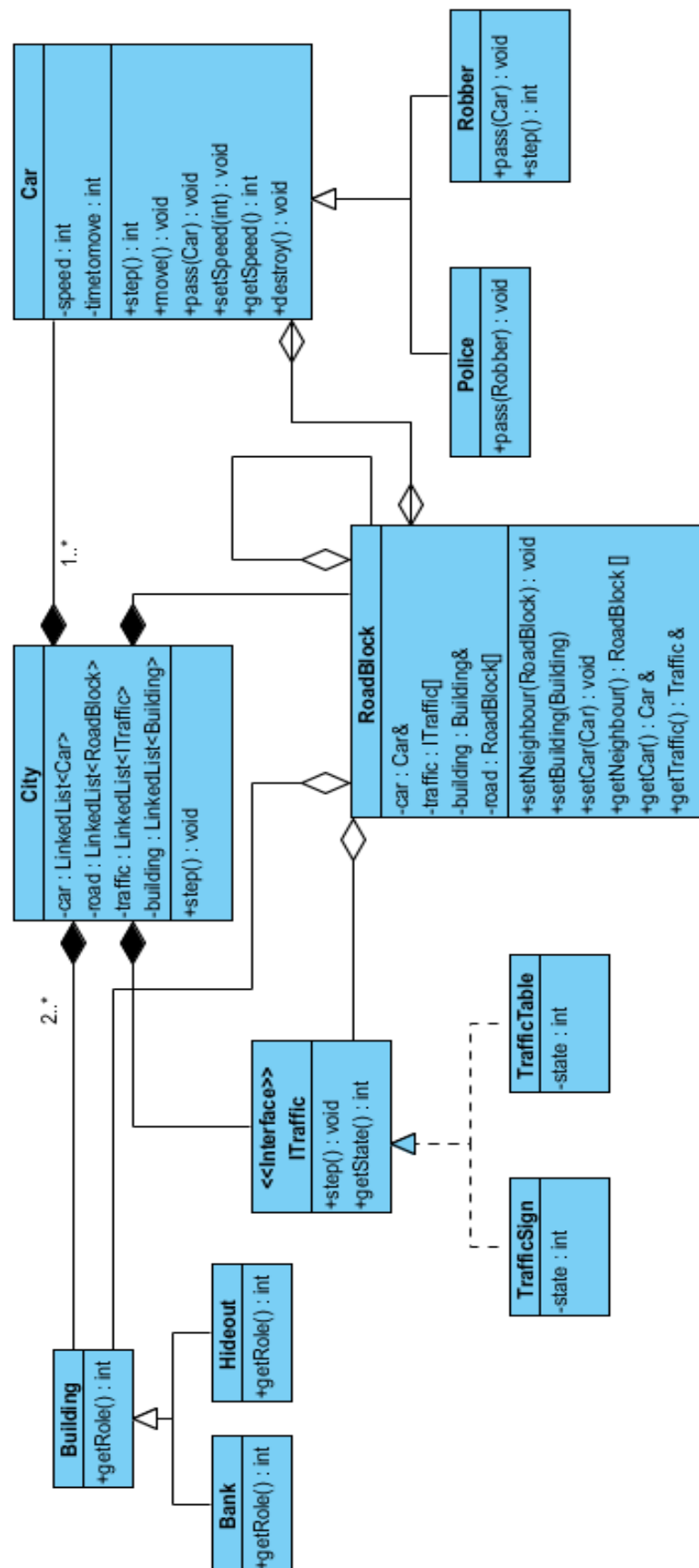
4.2.10 TrafficSign

- **Felelősség**
A jelzőlámpa, felelőssége az állapotának változtatása és annak közlése.
- **Ősosztályok**
nincs
- **Interfészek**
ITraffic
- **Attribútumok**
 - **int state** : aktuális állapota
- **Metódusok**

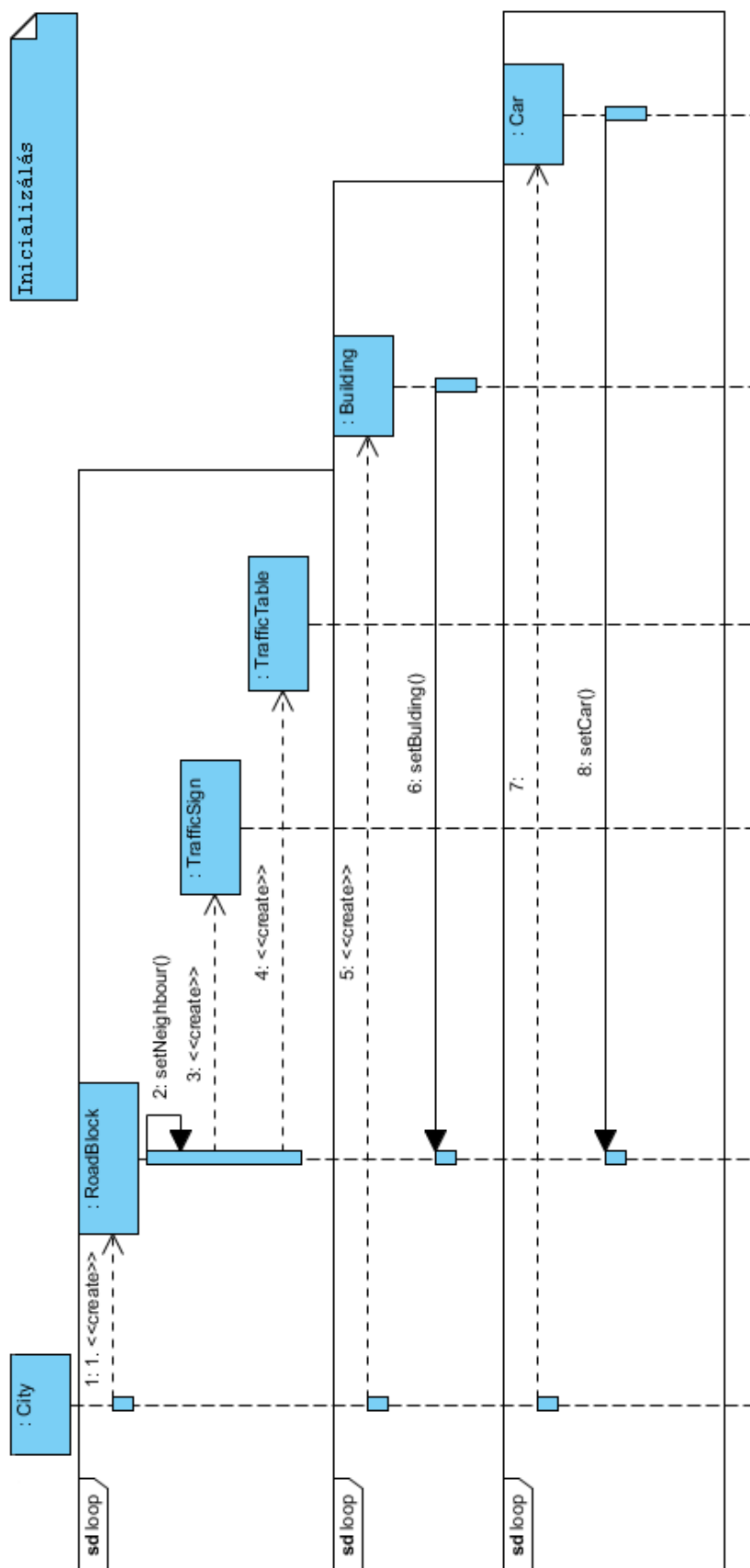
4.2.11 TrafficTable

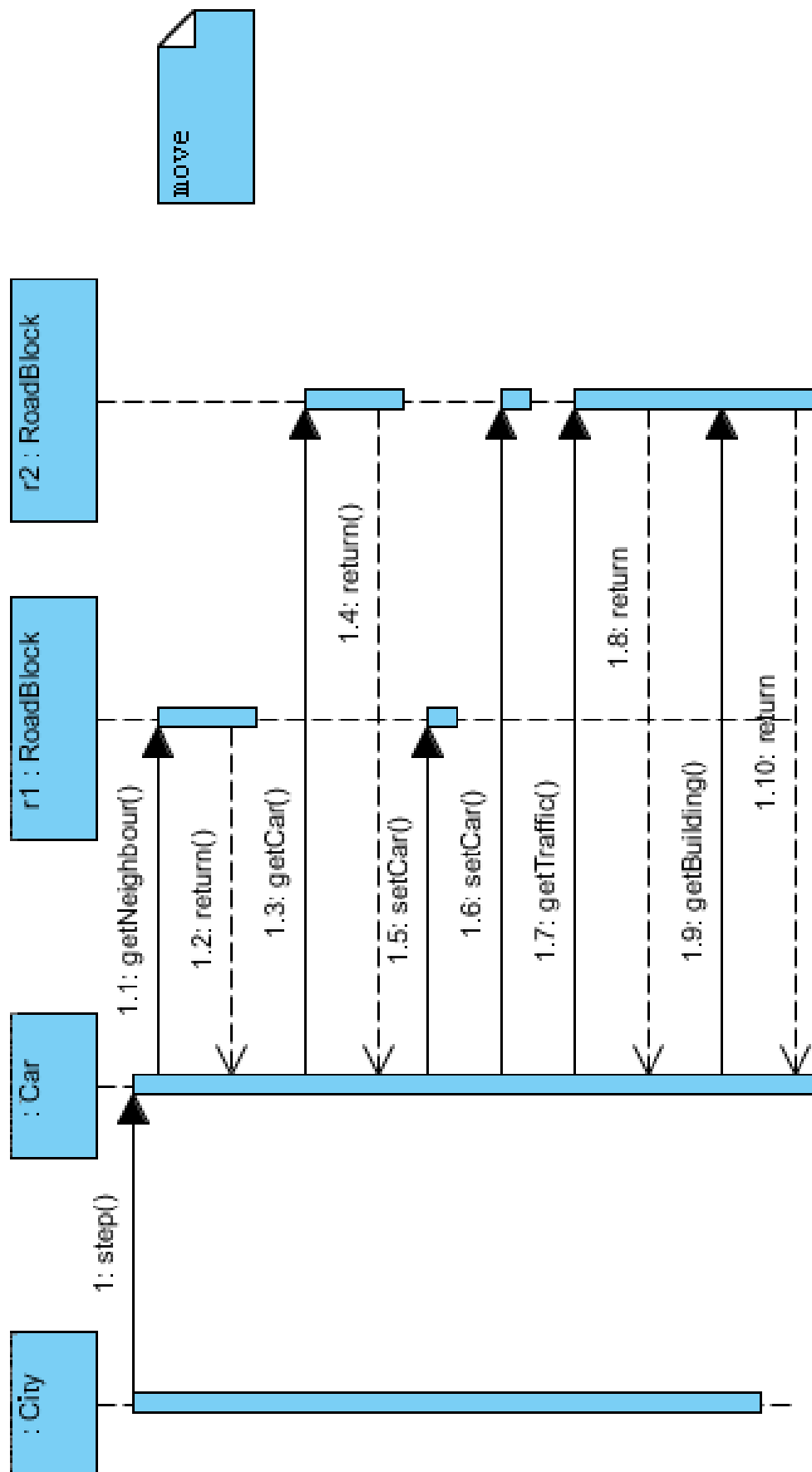
- **Felelősség**
Tábla, felelőssége a vonatkozó szabály közlése.
- **Ősosztályok**
nincs
- **Interfészek**
ITraffic
- **Attribútumok**
 - **int state** : a vonatkozó szabály
- **Metódusok**

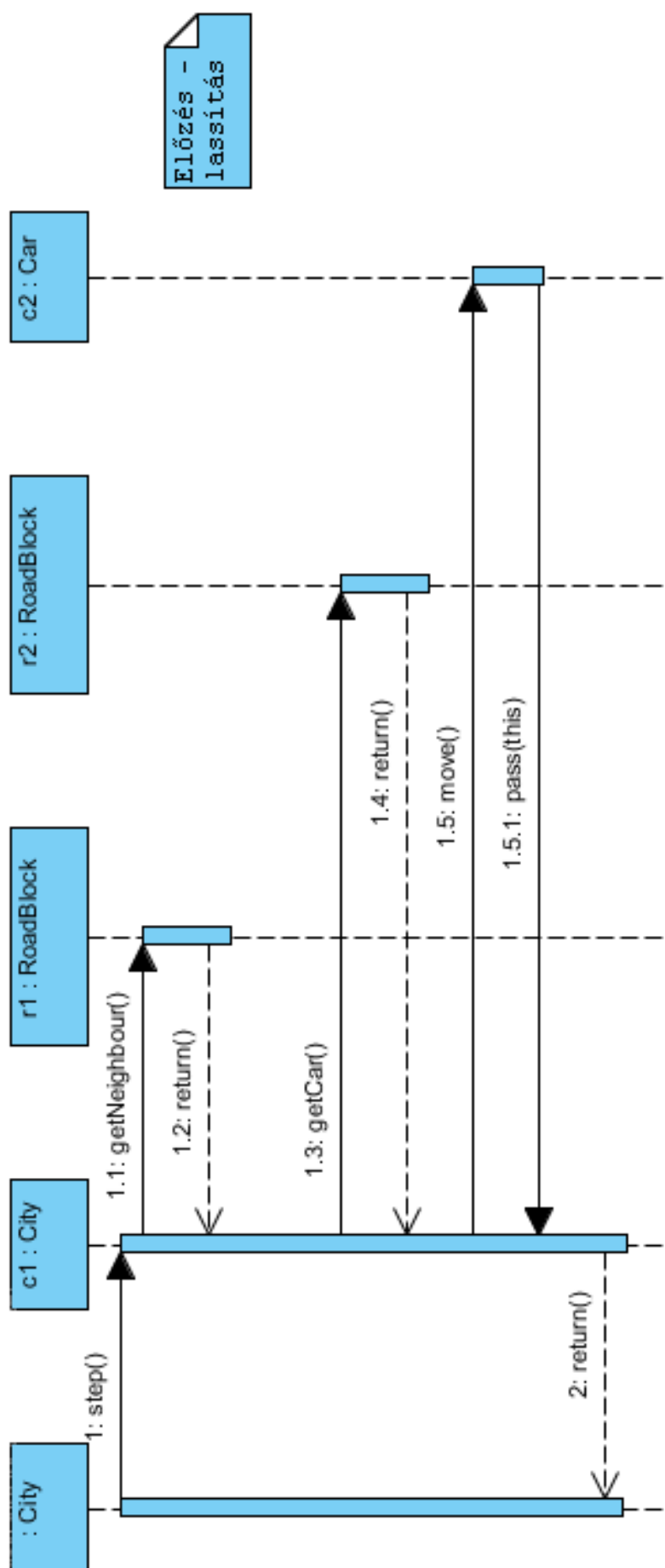
4.3 Statikus struktúra diagramok

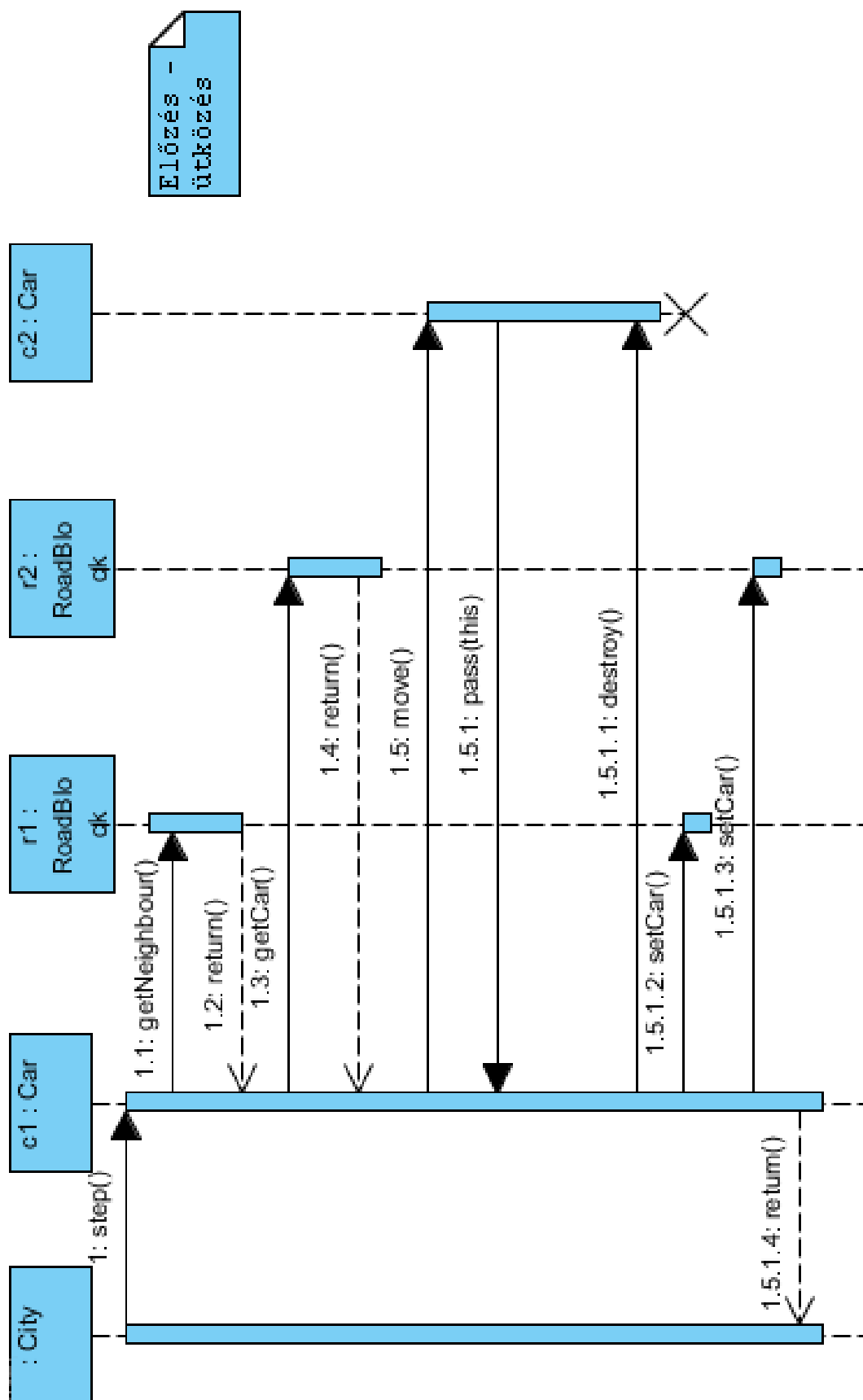


4.4 Szekvencia diagramok

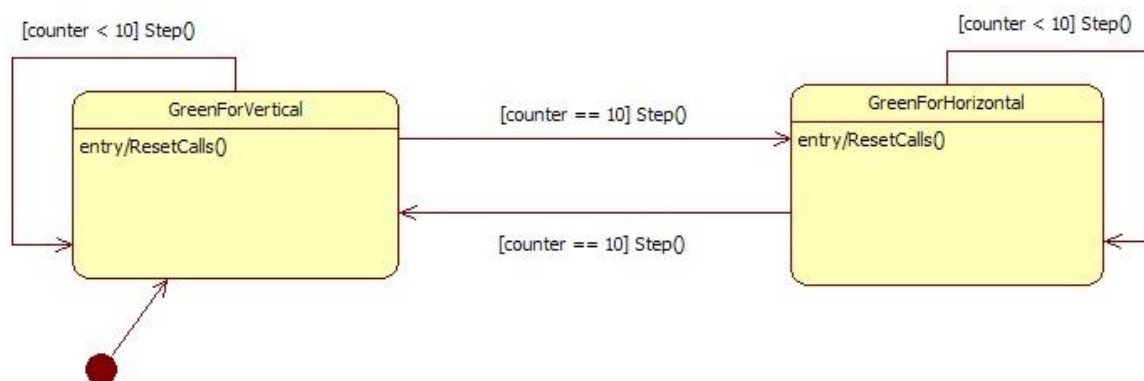




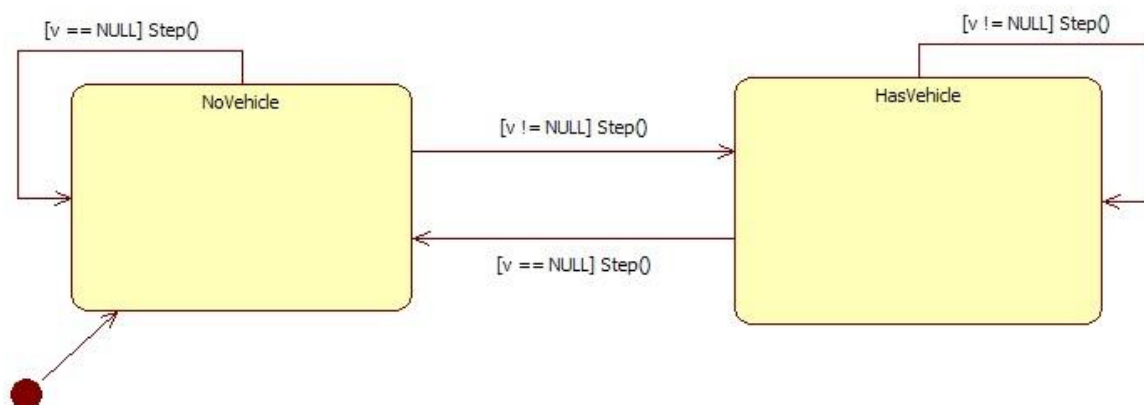




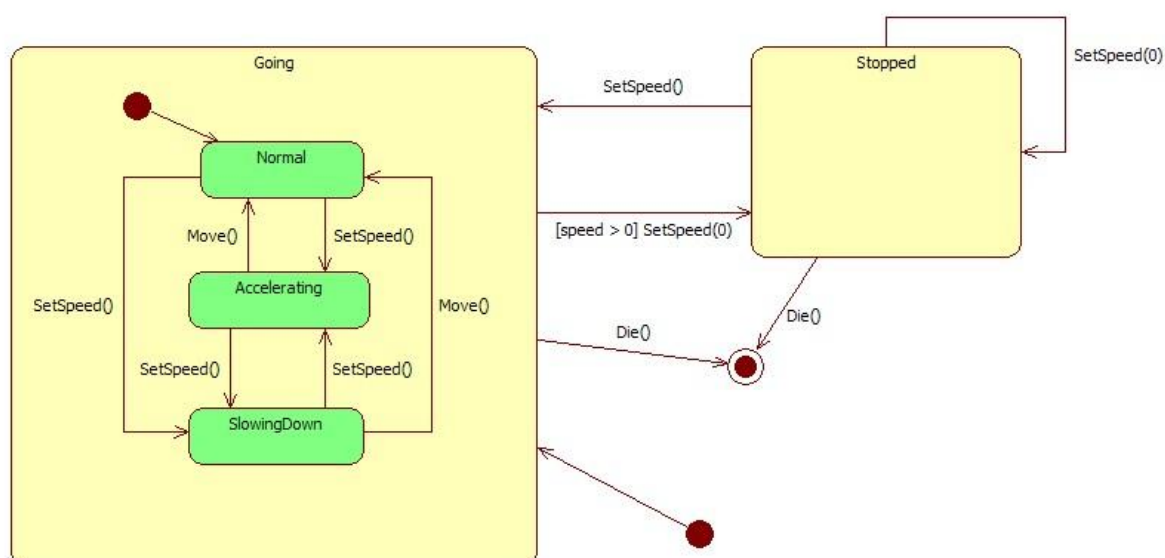
4.5 State-chartok



A jelzőlámpa állapotdiagramja



Az útelem állapotdiagramja



Az autó állapotdiagramja

4.6 Napló

Kezdet	Időtartam	Résztevők	Leírás
2010.03.01. 18:00	0,5 óra	Molnár Boros	Értekezlet. Döntés: Molnár elkészíti az szekvencia diagramokat és az objektumkatalógust, Boros az osztálydiagrammot, és az osztályok leírását.
2010.03.02. 19:00	3 óra	Boros	Elkészíti az osztálydiagrammot
2010.03.03. 08:00	1 óra	Molnár	Javítja az osztálydiagrammot
2010.03.03. 09:00	2.5 óra	Molnár	Elkészíti a szekvencia diagrammokat
2010.03.03. 20:00	2,5 óra	Boros	Elkészíti a az osztályok leírását
2010.03.03. 20:00	1.5 óra	Molnár	Elkészíti az objektumkatalógust
2010.03.03. 23:00	2 óra	Boros	Átszerkeszti, formázza az elkészült dokumentumot
2010.03.04. 9:00	0.5 óra	Rapp	Elkészült dokumentum ellenőrzése, gépelési hibák javítása