

# Intelligens Elosztott Rendszerek

## Házi feladat

*Banger csapat*

### Tagok:

csapatvezető: Seres Soma (LQH4X5)

Kiss Dániel Márk (WP871Q)

Róhringer Zoltán (QI91B2)

# Tartalomjegyzék

- 1) Feladateleírás
- 2) Megoldás összefoglalása
- 3) A fejlesztés összefoglalása
- 4) A program ismertetése
- 5) Bemutató videó

# Feladat leírása:

A feladatban egy okos éttermet valósítottunk meg, ami több ágensből épül fel. Az étteremnek a felhasználói felületen keresztül lehet rendelést leadni, ahol meg kell adni, hogy hova szeretnénk rendelni, illetve mit. Az étteremben egy ágens felel a készletek (alapanyagok az ételekhez) szinten tartásáért, egy ágens [szakács] az ételek készítéséért, egy ágens a nyersanyagok előkészítéséért, és a szakácshoz való eljuttatásáért[kukta], és egy ágens [pincér] az ételek kiszállításáért.

Az étterem célja az, hogy minél rövidebb idő alatt kiszolgáljon egyes rendeléseket, mivel minél gyorsabban kapja meg a rendelést a vevő, annál több vagyonra tesz szert, céljuk ennek a bevételnek a maximalizálása.

# A megoldás összefoglalása:

Amikor elindítjuk a programot, a pincér és a kukta elfoglalják a helyüket, és ilyenkor várják az új rendelést. Ezt egy külön ablakon lehet bevadni nekik, egy legördülő menüből kiválasztva. Három ételt közül választhatnak a vendégek, ezek a burgerek, spaghetti, és pizza. Ezen felül egy másik menüben ki lehet választani melyik asztalhoz ül le a vendég, vagyis melyik asztalhoz vigye ki a pincér a rendelést. Ez után a Rendelés gombot megnyomva leadhatjuk a rendelésünket.

A konyhában négy ágens található. Ezek egy supermarket, egy kukta, egy szakács, és egy pincér. Ezen felül, még van a konyhában egy hűtő (szürke kör a hűtő felirattal), meg három asztal (lila körök asztal 1, 2, 3 feliratokkal). Az ágenseknek szerepük szerint külön feladatuk van.

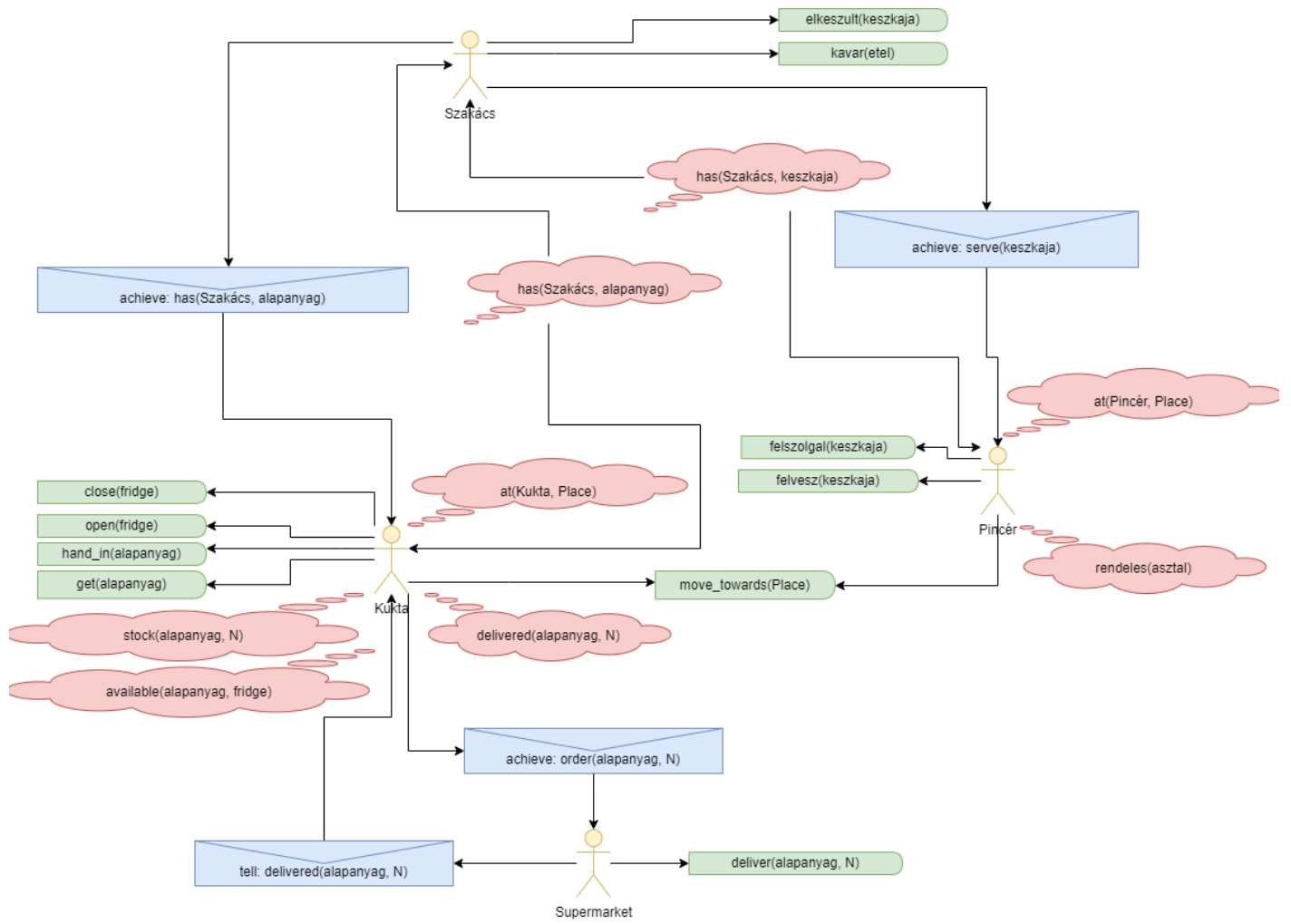
A **pálya**, ahol az ágensek mozognak egy 7x7-es rács, emellett még láthatók a statikus tárgyak, vagyis az asztalok, a hűtő, és egy statikus ágens, a szakács. Ezeken felül még van egy fal, ami elválasztja a konyhát, és az étkezőt, ezek a fekete négyzetekként vannak jelölve. Egy ajtón tud kimenni a pincér, ami csak egy sima mező.

A **kukta** (sárga kör, kukta felirattal) szállítja a nyersanyagot a szakácshoz, amelyet a hűtőből vesz ki. A különböző ételekhez különböző számú nyersanyag szükséges, a burgerhez egy, a pizzához kettő, a spaghettihez három. A kukta ezután elviszi a nyersanyagot a szakácshoz, ott átadja neki a nyersanyagokat, és együtt elkészítik azt. Miután végeztek a kukta visszatér a hűtőhöz, ahol várja az új rendelést, vagy, ha már időközben érkezett új, akkor már viszi is ahhoz az új nyersanyagokat a szakácsnak.

A **szakács** (szürke kör, szakács felirattal) feladata itt a nyersanyagokból az étel elkészítése. Akkor kezdődik ez a folyamat, amikor a kukta átadja neki a nyersanyagokat az ételhez. Ilyenkor elkezd az étel főzését. Akkor készül el az étel, amikor a szakácsnál a számláló nullát ér el. A számláló attól függ, hogy mennyi nyersanyag szükséges az étel elkészítéséhez. Miután a szakács befejezte a főzést, átadja a kész ételt a pincérnek.

A **pincér** (zöld kör, pincér felirattal) átveszi az ételt a szakácstól, és kiszállítja azt a megfelelő asztalhoz. Ott felszolgálja az ételt, és megkapja az ahhoz járó fizetséget, ami megegyezik az ételhez szükséges nyersanyag mennyiségével. Ha ez megtörtént, visszatér a szakácshoz, és várja az új kész ételt, hogy kivihesse az elkészült rendelést.

A **supermarket** ágens feladata, hogy újratöltse a hűtőt. A hűtőben a kezdetben öt nyersanyag található, ha ez a szám három alá megy akkor kell a supermarket ágensnek új anyaggal feltöltenie azt. Ilyenkor öt nyersanyagot vesz, és ezek egy dollárba kerülnek.



# A fejlesztés összefoglalása:

A fejlesztés során Jason 2.5.1-es verzióját használtuk. A Java fejlesztés során pedig JDK 18-as verzióját használtuk.

## Megvalósított funkciók ASL szinten:

- Ágensek definíciója és hiedelmeinek, céljainak felvétele
- Ágensek egymással való kommunikációja

## Megvalósított funkciók Java szinten:

- GUI felület a konyhához
  - Rendelés leadás felület
- Az ágensek mozgatása
- Útvonalkeresés
- A konyhára befolyt összeg nyilvántartása(+ bejövő összeg)
- Főzés során anyaghasználat(- nyersanyagmennyiség)

# A program ismertetése:

A programnak futás közben egy külön ablakon lehet megadni a rendeléseket, és az asztalt ahova leül a felhasználó. Egy legördülő menüben lehet kiválasztani a rendelést, és az asztalt is. Ezeket a program futása alatt bármikor meg lehet adni, és bármennyi rendelést is, a program meg fogja jegyezni mit adtunk be, és sorban ki fogja szolgálni a felhasználókat. Amint kiválasztottuk, a rendelés gombbal megrendelhetjük az ételt. Van még az interface-n egy számláló, ami az eddig megszerzett jövedelmet mutatja.

## **Supermarket:**

Beliefs:

-

Desires:

-

Intentions:

- `!order: true` -> a rendelés végrehajtása, feltölti a hűtőt alapanyaggal 1\$-ért cserébe.

## **Kukta:**

Beliefs:

- `available(alapanyag, fridge)` -> van-e alapanyag a hűtőben
- `at(kukta, P)` -> éppen P-nél tartózkodik-e a kukta
- `stock(alapanyag, N)` -> felméri a hűtő tartalmát, és ha kevesebb mint 3, akkor frissíti az elérhetőséget, ezáltal új adagot fog rendelni.
- `delivered(alapanyag, _Qtd): true` -> megérkezett a rendelés, frissíti az elérhetőséget, és folytatja a feladat lekérdezését.

Desires:

-

Intentions:

- `!has(szakacs, alapanyag): available(alapanyag, fridge)` -> el kell érje, hogy a szakácsnál legyen alapanyag, és van is a hűtőben. Ekkor elmegy a hűtőhöz, kinyitja, kiveszi a megfelelő mennyiségű alapanyagot, becsukja a hűtőt, elviszi a szakácsnak, átadja az alapanyag(ok)at és segít elkészíteni az ételt.
- `!has(szakacs, alapanyag) : not available(alapanyag, fridge)` -> el kell érje, hogy a szakácsnál legyen alapanyag, de nincs a hűtőben, ezért szól a supermarketnek, hogy érje el, hogy legyen 5-tel több, majd a hűtőhöz megy, hogy gyorsítsa a folyamat, és már ott várja a megérkezését az alapanyagoknak.
- `-!has(szakacs, alapanyag): true` -> folyamatosan ellenőrzi, hogy van-e a szakácsnak szüksége alapanyagra, ezzel mindig időben el tud indulni, amint kell.
- `!at(kukta, P): not at(kukta, P)` -> ha nincsen a céljánál, menjen az irányába
- `!at(kukta, P): at(kukta, P)` -> ha megérkezett egy adott helyre, akkor befejezi a mozgást.

## Pincér

### Beliefs:

- `rendeles(ures)` -> megadja, hogy hova kell a következő rendelést kiszállítani, ha nincs éppen elkészítendő rendelés, akkor "ures" értéke van.

### Desires:

- `!serve(keszkaja)` -> a pincér célja, hogy felszolgálja az elkészült ételt.

### Intentions:

- `+!serve(keszkaja)`: `has(szakacs, eszkaja)` -> ha van a szakácsnál elkészült étel, akkor fel kell szolgálni. Elmegy a szakácshoz (ha esetleg még nem lenne ott), felveszi tőle az ételt, megnézi, hogy melyik asztalhoz kell kivinni és odamegy, ott felszolgálja, majd visszamegy a szakácshoz, várva a további rendelésekre.
- `+!serve(keszkaja)`: `not has(szakacs, eszkaja)` -> ha nincs a szakácsnál elkészült étel, akkor is odamegy hozzá, hogy ezzel növelje a kiszolgálás gyorsaságát, hogy amint lesz, ne kelljen még oda is elsétálni, hanem már ott várja.
- `-!serve(keszkaja)`: `true` -> folyamatosan ellenőrzi, hogy kell-e rendelést felszolgáltatnia.
- `+!at(pincer,P)`: `not at(pincer, P)` -> ha nincsen a céljánál, menjen az irányába
- `+!at(pincer, P)`: `at(pincer,P)` -> ha megérkezett egy adott helyre, akkor befejezi a mozgást.

## Szakács:

### Beliefs:

- `has(szakacs, eszkaja)` -> van-e a szakácsnál kész étel.
  - ha van nála, akkor elküldi a pincért, hogy szolgálja fel
- `has(szakacs, alapanyag)` -> van-e a szakácsnál alapanyag.
  - ha van nála, akkor nekiáll főzni
  - ha nincs nála, akkor alapállapotba áll (pénzt akar szerezni)

### Desire:

- `!get(penz)` -> a szakács célja pénzt szerezni

### Intentions:

- `+!get(penz)` : `true` -> ahhoz hogy pénzt tudjunk szerezni, a kuktának alapanyagot kell hoznia.
- `+!cook(etel)` : `has(szakacs, alapanyag)` -> amíg van mit csinálni, kavargatja az ételt.
- `+!cook(etel)` : `not has(szakacs, alapanyag)` -> elkészült az étel.
- `-!cook(etel)`: `true` -> ha már nem kell főzni, akkor befejezi a főzést.

# Bemutató videó

<https://youtu.be/dxpbNY7KLVs>