



Programozó Döntő feladat
2014

Arany fokozatú támogatók



YAHOO!

Szakmai partnerek



Szervezők



BeeSmarter #2 programozói verseny

A verseny célja

Az idei programozói verseny feladata egy **app teljes megvalósítása**: egy alkalmazáskonceptió és párhuzamosan fejlődő dizájnerterv alapján kell a verseny végére a lehető legszebb és legjobb, a referencia tableten futó alkalmazást elkészíteni. Az általunk kiválasztott és előkészített applikáció egy fantasy kártyajátékon alapul, ahol a látvány kellőképpen érvényesülni tud, ugyanakkor maga a játék arra is lehetőség ad, hogy algoritmikus kihívással (mesterséges intelligencia építése) is szembesítse a csapatokat.

A **programozói versennyel szimbiózisban dizájnerversen** is zajlik, ahol a dizájnerversen csapatok az applikáció teljes dizájnertervén dolgoznak szoros együttműködésben a programozói csapatokkal. A mobil alkalmazások esetén kulcskérdés a felhasználói élmény (UX) maximalizálása, ami a dizájnerek és a programozók szoros együttműködését igényli. Ez a kapcsolat azonban nem mindig kiegyensúlyozott, a versenyen 24 órába sűrítettük azt a gyakran hónapokig tartó egyeztető folyamatot, ami végül a tökéletes felhasználói élményt nyújtó alkalmazást eredményezi, ezáltal növelve a verseny kihívásait.

A verseny lebonyolítása

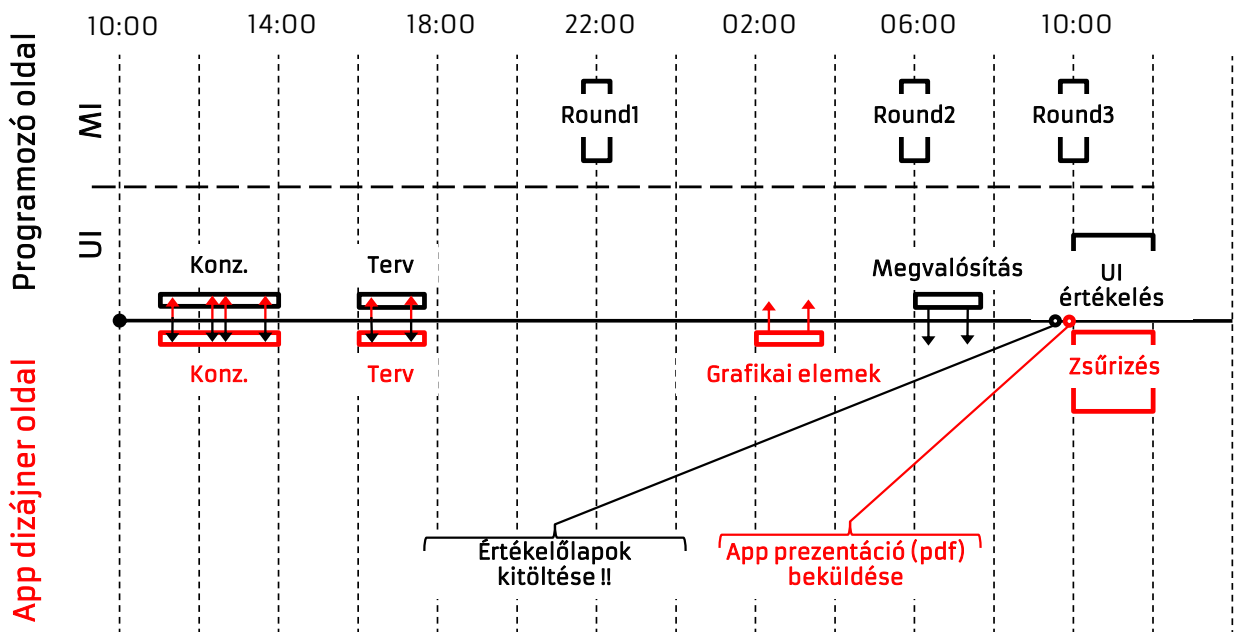
A verseny 24 órájában a feladat két szálon fut:

i) **Mesterséges intelligencia (MI) kifejlesztése**: feladat egy mesterséges intelligencia kifejlesztése, melynek hatékonyságát kettős kieséses rendszerben mérjük össze a többi robot intelligenciájával. A kettős kieséses bajnokságra három alkalommal (Round1, Round2, Round3) kerül sor, emelkedő megszerezhető pontszámmal. Természetesen a mesterséges intelligencia egészen a verseny végéig fejleszthető.

ii) **Az UI megvalósítása**: A verseny során két programozói csapat mellett lesz egy dizájnerversen csapat. A dizájnerversen csapat feladata az alkalmazás UI koncepciójának kialakítása szoros egyeztetésben a programozó csapatokkal külön-külön. Az együttműködés sikeressége is értékelési szempont (a részleteket az "Értékelési szempontok" szakaszban írtuk le)! A programozók feladata a dizájnerversen csapattal konszenzusos alapon kialakított UI megvalósítása teljes látványában, amihez a dizájnerek biztosítják a szükséges grafikai elemeket. A programozók esetén a megvalósítást pontozzuk (nehézség szerint súlyozva), nem magát az UI-t (a részleteket az "Értékelési szempontok" szakaszban írtuk le)! Az alkalmazás UI fejlesztése a

24 órás mobil programozói verseny feladatai verseny során folyamatos, a végleges applikációt a kiadott tabletre telepítve kell bemutatni az értékeléshez/pontozáshoz.

A verseny javasolt ütemterve a következő ábrán látható.



A verseny megkezdése utáni rövid értelmezési időszak után a dizájnerek csapat felkeresi a számukra kijelölt két programozó csapatot (**Konz.**), és konzultálnak az alkalmazás alapkonceptjéről és a megvalósítandó (ill. a 24 órában megvalósítható) UI tervekről. A dizájnerek egy újabb megbeszélésen (**Terv**) bemutatják az előzetes terveket, amelyek közül konszenzusos alapon kiválasztásra kerül a végleges terv. A dizájnerek az UI-hoz szükséges kész grafikai elemeket a verseny második szakaszában (**Grafikai elemek**) adják át a programozó csapatoknak. A programozó csapatok ezeket behúzzák, és a tervek alapján elkészítik az alkalmazás első változatát, amit bemutatnak a dizájnerek csapatnak (**Megvalósítás**). Az alkalmazás UI oldala ezek után szükség szerint még persze finomítható.

Az UI fejlesztésével párhuzamosan a programozó csapatok az MI fejlesztésén is dolgoznak, melyet a verseny során három fordulóban neveznek versenyre. A fordulók lebonyolítása során a csapat (legalább) egy játékosának a versenytérben (Aula) kell megjelennie a forduló kezdete előtt 5 perccel, kezében a tabletre telepített alkalmazással. Két MI közötti meccs során az nyer, aki 3 játszma lejátszásából több játékot nyer. A forduló kettős kieséses rendszer szerint szerveződik, a csapatok helyezése szerint kapnak pontokat (a részleteket az Értékelési szempontok szakaszban írtuk le). A **Round1** és **Round2** esetén az UI nem lényeges (csak minimális GUI szükséges), ugyanakkor a **Round3** fordulóban a végleges alkalmazásnak kell futnia a tableten. Ez a záróforduló már nagyobb

24 órás mobil programozói verseny feladatai nyilvánosságot kap. Az applikációk UI értékelése is ebben az időszakban történik (**UI értékelés**), a részpontszámok összegzése után alakul ki a verseny végső rangsorra.

Értékelési szempontok

A verseny értékelésekor három szempont érvényesül: i) az **MI** komplexitása, ereje; ii) a megvalósított **UI** szépsége, kidolgozottsága (itt nem a dizájn kidolgozottságára gondolunk, hanem az implementációra) ill. **UX**; iii) együttműködés a dizájnner csapattal. Mivel az első két szempont egymástól független, ezért két szélső stratégia (csak UI megvalósítás, MI fejlesztés nélkül vagy csak MI fejlesztés, UI megvalósítás nélkül) között a programozó csapatok szabadon választhatnak pontszerzési stratégiát, és eszerint oszthatják be az értékes idejüket (ami véges). Az értékelési szempontokat és az elérhető pontszámokat az alábbi táblázat foglalja össze:

Értékelési szempont	Rövid leírás	Mikor? Hogyan?	Elérhető pontszám
UI (GUI)	Megvalósítás, animáció, effektek, layout együttese.	vasárnap 10:00, zsűri	350 pont
UX (audio)	Hangeffektek, háttérzene együttese	vasárnap 10:00, zsűri	100 pont
UX (app)	Vizuális hierarchia, gördülékenység, kreatív megoldások együttese	vasárnap 10:00, zsűri	100 pont
MI (Round1)	MI hatékonyság mérése (3 játszma/ mérkőzés)	szombat 22:00, tournament	max. 50 pont
MI (Round2)	MI hatékonyság mérése (3 játszma/ mérkőzés)	vasárnap 06:00, tournament	max. 100 pont
MI (Round3)	MI hatékonyság mérése (1 játszma/ mérkőzés)	vasárnap 10:00, tournament	max. 150 pont
Dizájnner értékelés	A programozók együttműködési és kommunikációs képességeit méri dizájnner szempontból.	vasárnap 10:00, dizájnner űrlapot töltve adják	50 pont
		Összesen:	900 pont

Az egyes értékelési szempontok részletes kibontása a következő:

- **UI (GUI):**

Szempont	Leírás	Elérhető pontszám
Megvalósítás	A designer tervekhez való hűség	120
Animációk, effektek	Kártyák mozgatása, játék állapotváltozások szemléltetése	110
Layout	Átláthatóság, UI elemek elrendezése	120

- **UX (non-vizuális):**

Szempont	Leírás	Elérhető pontszám
Hangeffektek	A játékeseményeket kísérő hangok	50
Hattérzene	A játék hangulata a játék különböző állapotaiban	50

- **UX (app):**

Szempont	Leírás	Elérhető pontszám
Vizuális hierarchia	Felépítés, funkciók elérhetősége	35
Gördülékenység	A felhasználói élmény folytonosságát nem akasztja meg semmi	35
Kreatív megoldások	Egyedi interakciók	30

- **MI (Round1-3):** a fordulók helyszíne az Aula, ahol a szervezők irányítása mellett bonyolítjuk a kettős kieséses rendszerű körmérkőzéseket. A konkrét szabályokat a játéklevezető szervezők a játszmák megkezdése előtt ismertetik. A helyezéstől függően fordulónként az alábbi pontszámok szerezhetőek meg:

Helyezés	Round1	Round2	Round3
1.	50 pont	100 pont	150 pont
2.	40 pont	80 pont	120 pont
3.	30 pont	60 pont	90 pont
4.	25 pont	50 pont	75 pont
5.	20 pont	40 pont	60 pont

24 órás mobil programozói verseny feladatai

6.	18 pont	36 pont	54 pont
7.	16 pont	32 pont	48 pont
8.	14 pont	28 pont	42 pont
9.	12 pont	24 pont	36 pont
10.	10 pont	20 pont	30 pont
11.	8 pont	16 pont	24 pont
12.	6 pont	12 pont	18 pont
13.	4 pont	8 pont	12 pont
14.	2 pont	4 pont	6 pont

Az első illetve második helyezett a forduló döntőjének győztese illetve vesztese, a további helyezések a megnyert játékok számának függvényében lesznek kiosztva. Ha egy csapat nem jelenik meg a fordulón, akkor automatikusan az adott fordulón elért pontszáma 0 pont. Ha egy csapat applikációja játék közben kifagy, akkor a játékfelügyelők ezt regisztrálják, a második kifagyásnál az adott játszmát elvesztette. (Az egyes fordulóknban egy mérkőzés 3 játszmából áll.)

- **Dizájner értékelés:** a versenyben hangsúlyos szerepe van az app dizájner-programozó interakciónak, aminek programozói oldalát a dizájner csapatoktól kapott értékeléssel pontozzuk. A dizájnerek értékelőlapján 4 kategória szerepel: i) pontos megfogalmazás, érthetőség (0-10 pont), ii) segítőkészség (0-10 pont), iii) együttműködési hajlandóság (0-10 pont), iv) megvalósított UI tervhűsége (0-20 pont).

A dizájner értékeléshez hasonlóan a programozók is értékelik a dizájnereket ugyanolyan együttműködési és kommunikációs szempontok szerint.

A feladat

A feladat egy körökre osztott online fantasy kártyajáték mobil kliensének megvalósítása.

A kliens feladata, hogy a játék szerverhez csatlakozva a számára kiosztott ellenféllel egy csatát a játékszabályok szerint megvívjon és megnyerjen.

A kliens egy játékszerverhez kapcsolódik, aminek API hívásaival végre tudja hajtani a csata lebonyolításához szükséges lépéseket. Ezeket részletesen a **Játékszabályok** fejezet írja le.

A lépésekről a kliens mesterséges intelligenciája dönt. Az ehhez rendelkezésre álló döntési idő korlátozott, túl hosszú gondolkozási idő esetén a kliens automatikusan elveszíti az aktuális játszmát.

A kliens feladata továbbá, hogy a csata aktuális állását a felhasználói felületen megjelenítse. Az alkalmazásnak biztosítania kell a manuális játék lehetőségét is, amelyhez a lehető legjobb felhasználói élményt kell nyújtania.

Játékszabályok

Az alábbi fantasy megfogalmazású játékszabályok a játéklogika könnyebb megértéséhez adnak támaszt. A dizájnerek ehhez a játéklogikához teljesen más vizuális koncepciót (skin) is tervezhetnek, ugyanakkor a játéklogika és a paraméterek nem módosulhatnak.

A játékban egy varázsló szerepébe bújhatsz, aki más varázslókkal küzd meg lények és varázslatok megidézésével. A varázsló 5 mágikus elemet használ a csata során: Tűz, Víz, Levegő, Föld és Illúzió. Minden mágikus elemhez tartozik 4 kártya, tehát a varázsló az előtte fekvő deckből válogathat kártyalapokat, a deck mérete 5 x 4.

Minden varázsló 60 életerő ponttal kezdi a játékot. Minden varázsló rendelkezik további 5 számmal, amelyek a varázsló erejét reprezentálják az 5 mágikus elemre vonatkozóan. Minden alkalommal, amikor egy lényt vagy varázslatot megidézel, a megfelelő elemhez tartozó erő az idézés költségével fog csökkenni (ha nincs elég erőforrás, akkor a kártya nem játszható ki). Ugyanakkor ezek minden körben egy ponttal nőnek.

Minden megidézett lény rendelkezik támadó erővel és életerővel. Ha egy lény megtámad egy ellenséges lényt, akkor az ellenséges lény életerője a támadó lény támadó erejével csökken. Ha egy lény életerője elfogy, akkor a lény meghal.

Minden varázsló 6 megidézett lényel rendelkezhet egyszerre, ehhez 6 slot áll rendelkezésre a képernyőn. Minden körben a már megidézett lények támadják az ellenfelet. Minden lény külön támad, ilyenkor az ellenfél megidézett lényei az ellenfél első védvonalaként viselkednek: a támadó lények a velük szemközt

24 órás mobil programozói verseny feladatai slotban védekező lényeket sebzik. Ha egy támadó lényel szemközti slotban nincs védekező lény, akkor a támadó közvetlenül az ellenséges varázslót sebzí, annak életerejét a támadás erejével csökkentve.

A varázslatok olyan speciális kártyák, amik a kijátszás során a következő hatásokat válthatják ki:

- közvetlen sebzés egy célpontra: lényre vagy játékosra (*directDamage*),
- sebzés az ellenfél összes lényén (*damageAllCreature*),
- közvetlen sebzés az ellenfél varázslón (*directDamage*),
- egy saját megidézett lény gyógyítása (*heal*),
- az összes saját megidézett lény gyógyítása (*healAllCreature*),
- a varázsló gyógyítása (*heal*),
- egy elemi erő növelése (*resourceAdd*).

A játék események - pl.: lény idézés, varázslat megjátszása - kiválhatnak módosító hatásokat a fentiek alapján. Ezek közül egyszerre több is elsülhet.

Egyes lények szintén rendelkeznek a fenti speciális hatásokkal, ám szemben a varázslatokkal ezek nem csak kijátszás pillantában lehetnek aktívak:

- hatás csak a megidézéskor (*onSummon*)
- hatás minden körben (*onSummonOVT*)

A játszmat akkor nyered meg, ha az ellenséges varázsló életereje a te körödben elfogy.

Ha valami nem világos, akkor a tájékozódáshoz (a játék könnyebb megértéséhez) a kibővített játékról az alábbi linken találtok videót: <http://www.spectromancer.com/>

A kártyák

A játékban mágikus elemenként 4 kártya érhető el (összesen tehát $5 \times 4 = 20$ kártya van implementálva a szerveren). A játékban a teljes kártyapakli rákerül a deckre.

A Tűz mágikus elem kártyái a következők:

Kártya neve	Típus	Költség	Támadóerő	Életerő	Jellemző
Wall of Fire	lény	2	0	5	When Fire Wall is summoned it deals 5 damage to opponent's creatures.
Priest of Fire	lény	3	3	13	Priest of Fire increases by 1 the growth of owner's Fire power.
Bargul	lény	8	8	22	When Bargul is summoned it deals 4 damage to each other creature.
Inferno	varázslat	9	-	-	Deals 18 damage to all opponent's creatures.

A **Víz** mágikus elem kártyái a következők:

Kártya neve	Típus	Költség	Támadóerő	Életerő	Jellemző
Sea Sprite	lény	2	5	22	Sea Sprite deals 2 damage to owner each turn.
Merfolk Apostate	lény	3	3	10	When Merfolk Apostate is summoned it increases owner's Fire power by 2.
Mind Master	lény	8	6	22	Mind Master increases by 1 the growth of all owner's powers.
Astral Guard	lény	10	1	16	Astral Guard decreases by 1 the growth of all opponent's powers.

A **Levegő** mágikus elem kártyái a következők:

Kártya neve	Típus	Költség	Támadóerő	Életerő	Jellemző
Faerie Sage	lény	4	4	19	When Faerie Sage is summoned it heals 4 life to it's owner.
Phoenix	lény	5	5	18	When Phoenix is summoned it decreases the opponent player's Fire power by 3.
Chain Lightning	varázslat	8	-	-	Deals 9 damage to opponent and each of opponent's creatures.
Tornado	varázslat	10	-	-	Destroys target opponent's creature. (100dmg)

A **Föld** mágikus elem kártyái a következők:

Kártya neve	Típus	Költség	Támadóerő	Életerő	Jellemző
Elven Healer	lény	1	2	12	Elven Healer heals owner 3 life each turn.
Rejuvenation	varázslat	4	-	-	Caster gains 10 life.
Stone Rain	varázslat	9	-	-	Deals 25 damage to each creature.
Hydra	lény	10	3	40	At the beginning of it's owner's turn Hydra deals 3 damage to each of opponent's creatures.

A **Illúzió** mágikus elem kártyái a következők:

24 órás mobil programozói verseny feladatai

Kártya neve	Típus	Költség	Támadóerő	Életerő	Jellemző
Dream of Plenty	varázslat	1	-	-	Increases owner's Fire, Air, Water Earth, powers by 1.
Wall of Illusions	lény	5	0	32	Wall of Illusions deal 3 damage to opponent each turn.
Spectral Assassin	lény	5	6	24	When Spectral Assassin is summoned it deals 12 damage to opponent.
Infinite Wall	lény	5	0	60	-

A játékhoz tartozó kártyákról, az alábbi linken találtak leírásokat:

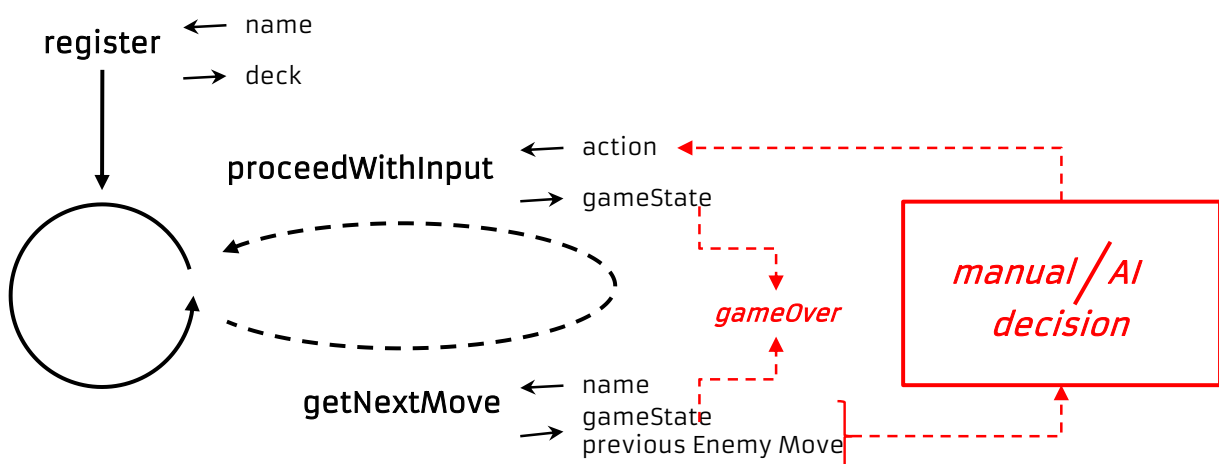
http://www.spectromancer.com/cardlist_en.htm

Kommunikáció a szerverrel

A játékot egy REST API segítségével vezérelhetik a játékosok, az alábbi hívásokkal:

- *registerPlayer*: beeregisztrál egy játékost (Szerver felé),
- *getNextMove*: a következő lépést engedélyét megkezdését kérdezi le,
- *proceedWithInput*: a választott lépésünkről közli (Szerver felé),
- *restartGame*: új játékot indít (Szerver felé),
- *resetServer*: kitörli az eddig regisztrált játékosokat és a megkezdett játékot.

A szerverhívások logikai kapcsolatát a következő folyamatábra szemlélteti.



Minden kérelemre az adott formában érkezik válasz:

```
{
  "success": true/false,
  "error": [ "fail reason 0", "fail reason 1" ],
  "result": { "someResult": "forQuery", "or": "empty" }
}
```

Hiba esetén értelemszerűen a "success" értéke false, és az error tömb tartalmaz egy rövid leírást a hiba okáról.

A játékos (Player) regisztrálása a következő hívással történik:

host/registerPlayer?name=someNiceName

A már zajló játékba nem lehet regisztrálni. A játékosok egyedi névvel kell, hogy regisztráljanak, később ezzel a névvel azonosítják magukat. Sikeres regisztráció után a játékos megkapja a paklija tartalmát.

method	path	parameters	name
GET	/registerPlayer	-	a játékos neve

A hívásra adott válasz az alábbi:

```
{
  "cards":
  {
    "fire": [ { card0, card1, card2, card3 } ],
    "water": [ { card0, card1, card2, card3 } ],
    "air": [ { card0, card1, card2, card3 } ],
    "earth": [ { card0, card1, card2, card3 } ],
    "illusion": [ { card0, card1, card2, card3 } ]
  }
}
```

A játékos a következő lépésről az alábbi hívással értesül:

host/getNextMove?name=someNiceName

Ennek a kérésnek a folyamatos hívásával (polling) értesülhet a játékos az ellenfél döntéséről és vice-versa. A válasz "proceed" értéke igaz, ha a játékos következik, az "enemyInput" írja le az ellenfél döntését, és a "gameState" az aktuális állapotteret.

method	path	parameters	name
GET	/getNextMove	-	a játékos neve

A hívásra adott válasz az alábbi:

```
{
  "proceed": true,
  "enemyInput": { "action": "skipTurn" },
  "gameState": { "currentPlayerIndex": 1, "gameOver": false, "players": [ ... ]
}
```

A játékos lépése a következő hívással érvényesül:

host/proceedWithInput?name=someNiceName&action=playCard&resourceType=fire&cardIndex=2&slotIndex=4

Három lehetséges "action" ("playCard", "skipTurn", "surrender") közül kell egyet választani. A kártya kijátszásakor szükség van a kártya típusára ("fire", "water", "air", "ice", "illusion") és indexére (0..3). Ha a kártya kijátszásához további adatra van szükség, akkor az a slotIndex reprezentálja (például lény kártya kijátszásakor a slotot, amibe a lényt rakjuk azonosítani kell).

method	path	parameters	name
GET	/proceedWithInput	action: a játékos döntése ("playCard", "skipTurn", "surrender") resourceType: a kártya erőforrás típusa cardIndex: a kártya indexe (0..3) slotIndex: a cél slot (0..5)	a játékos neve

A játék újraindítása a következő hívással történik:

host/restartGame

A regisztrált játékosok és a kártyák nem törlődnek. Verseny alatt nem hívható!

method	path	parameters	name
GET	/restartGame	-	-

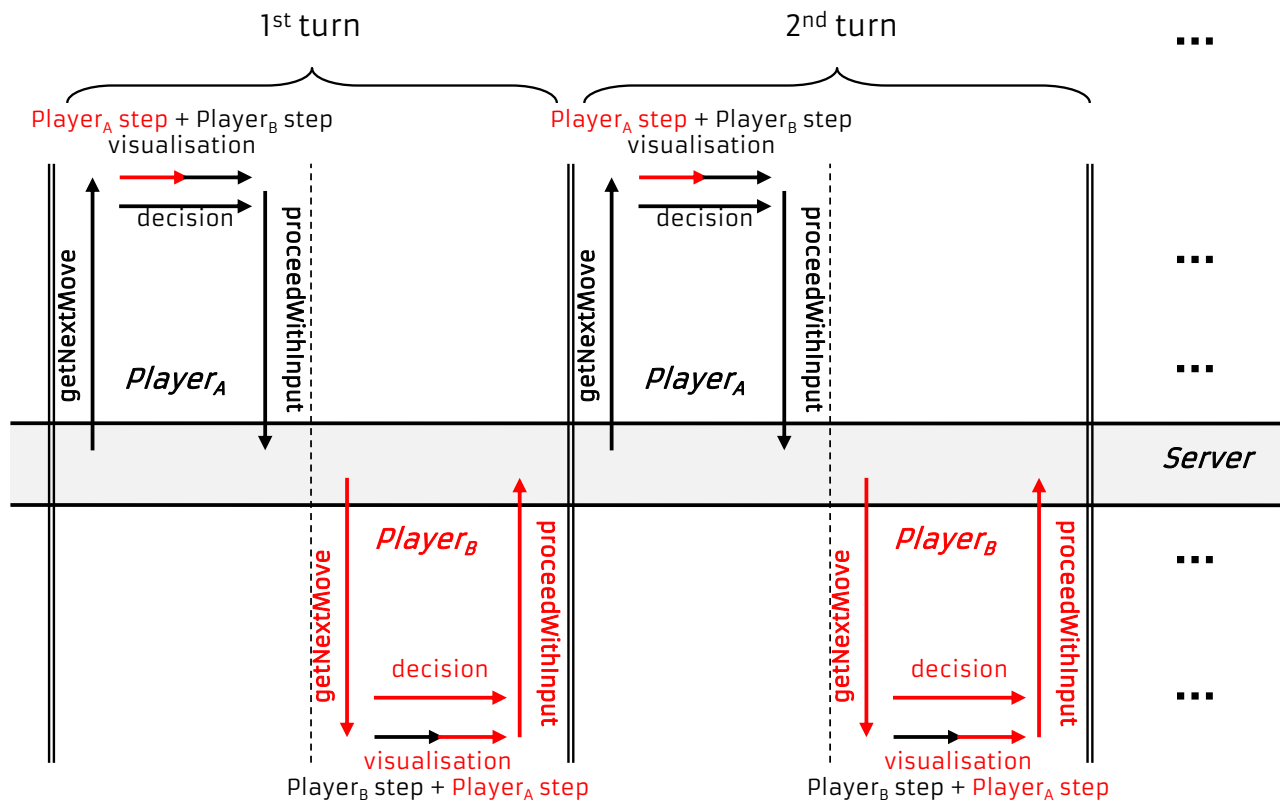
A szerver újraindítása a következő hívással történik:

host/resetServer

A teljes szerver reset a regisztrált játékosokat törli.

method	path	parameters	name
GET	/resetServer	-	-

A játék akkor indul, ha két eltérő nevű játékos regisztrál. Az elsőként regisztráló kezdi a játékot. A játék során a játékosok felváltva hívják a szervert, ahogy azt a következő ábra is mutatja.



A szerver hívások között a kliens feladata a döntés (manual vagy AI) és a vizualizáció (először az ellenfél, majd a döntés után a saját lépés) megvalósítása.

FONTOS: A szerver lokális futtatásához a következő helyen érhetőek el a szükséges fileok: http://beesmarter.jumpnet.eu/bs2_server.zip

Felhasználói felület

A felhasználói felülettel a következő funkciókat kell elérhetővé tenni:

- játék indítása / játék megállítása (kilépés a játékból),
- váltás manuális (*manual*) és MI játékos között (*AI*),
- játék állapotának szemléltetése (akár hangokkal, animációkkal),
- manuális játszhatóság (kijátszható kártyák, pontok, slotok, az ellenfél slotjai láthatóak).