

# Szöveges kalandjáték értelmező felhasználói dokumentáció

Bánáti Benedek

November 26, 2022

## 1 Értelmező működése

A program egy szöveges kalandjáték megjelenítéséért és vezérléséért (betöltés, mentés, játék beli állapot változások) felelős. A felhasználó által megadott logika alapján fog működni a játék.

### 1.1 Két fájlra osztott értelmezés

Két fájl van, amit a játék különböző működéséhez módosítani lehet. `objects.stry` és `rooms.stry`. Az első fájlban a szobában elhelyezett objecteket lehet módosítani/újakat alkotni, míg a másodikban a szobákat. A játékban minden ami megjelenik vagy egy object vagy egy room. Az objectek a roomokban helyezkednek el. A szobákban csak tartózkodni lehet, nem lehet velük interraktálni (kivéve mozogni köztük). Az objectekkel lehet interraktálni, amik pedig akár szobák akár objectek állapotát tudják változtatni. A kezdetleges állapota egy szobának vagy objectnek mindig az lesz amelyiket legelőször definiáltuk a fájlokban. A két fájl felépítése:

#### 1.1.1 `rooms.stry`

**id** : Azonosító, hogy egy szobának lehessen több állapota ami ugyan ahhoz a szobához tartozik.

**state** : Egy szobát megkülönböztető állapotok. Tehát egy szobából létezhet több változat, ilyenkor az állapotuk különbözteti meg őket. A 0-s állapot a deaktiváltat jelenti, tehát nem jelenhet meg a világban, a >0 állapotok pedig különböző, aktivált állapotok.

**name** : Egy szoba neve, amire a `go to/enter` parancs értelmeződni fog.

**look around text** : Ez lesz az a szöveg amit a játék kiír ha a játékos a "look around" parancsot használja.

**entry text** : Ez lesz az a szöveg amit a játék kiír ha a játékos belép ebbe a szobába.

**connected items** : ", "-vel elválasztott id-k, ami megmondja, hogy egy szoba milyen objecteket tartalmaz. Az id-knak létezniük kell az objects.stry id oszlopában.

**connected rooms** : ", "-vel elválasztott id-k, ami megmondja, hogy egy szoba milyen másik szobákhoz kapcsolódik. Az id-knak létezniük kell a rooms.stry id oszlopában.

### 1.1.2 objects.stry

Ez a fájl fogja tartalmazni a játékban megjelenő összes eszközt, azaz objectet és ezeknek a környezetre történő behatásukat. Képesek modify eventek alapján más objectet vagy vagy szobák állapotát változtatni. Egy objectnek a következő mezőkkel kell rendelkeznie.

**id** : Object azonosítója

**state** : Egy objectet megkülönböztető állapotok. Egy objectből létezhet több változat, ilyenkor az állapotuk különbözteti meg őket. A 0-s állapot a deaktiváltat jelenti, ilyenkor az object nem érhető el a világban, viszont a >0 állapotok különböző állapotait jelzik az objectnek.

**name** : Ez lesz az object neve. A játékos ezzel a szöveggel fogja tudni elérni azt az objectet.

**look at text** : Ez az a szöveg amit kiír a játék ha a játékos a "look at <object>" parancsot használja

**collectible** : "true" és "false" értéket vehet fel. Azt jelzi, hogy az adott objectet a játékos fel tudja-e venni és az eszköztárába helyezni.

**modify event** : ", "-vel elválasztott események listája. Ha valamelyik sorszámú esemény teljesül, akkor a modify mező azonos sorszámú eseményeit fogja teljesíteni. 3 Ilyen esemény felismerésére képes a program. Ha az objectet használni akarjuk, azaz a "use <object>" parancsot kapja a felhasználótól, ilyenkor a "use" eventet kell használnunk. Ha valamit használni akarunk az objecttel, azaz "use <object1> on <object2>" parancsot kapja a program, akkor a "use <object1>" eseménnyel tudjuk ezt definiálni. Ilyenkor az <object1> eszköznek benne kell lennie a játékos eszköztárában. Ha bármilyen más szöveget írunk be, az esemény akkor fog teljesülni ha a pontos szöveget kapja a felhasználótól. Még üresen is hagyhatjuk a mezőt, és ilyenkor ha a játékos megpróbálja használni az eszközt, akkor a "You can't use this item right now" üzenet fog megjelenni. "/" jellel választhatjuk el ha esetleg több esemény egyértelmű. A parancsokból a kötőszavakat a program automatikusan el fogja hagyni, ahogy a fölösleges whitespace karaktereket is.

Példa: open the door/use

A fenti példa akkor fog igazra teljesülni, ha a játékos azt a parancsot adja,

hogy "open the door" vagy "use door" (feltéve, hogy az object neve az, hogy door)

Példa 2: use 4

Csak akkor fog igazra teljesülni, ha a 4-es azonosítóval rendelkező object benne van a játékos eszköztárában, és a játékos azt a parancsot adja ki például, hogy: "use key on door" (feltéve, hogy a key object azonosítója 4)

**modify** "," karakterrel elválasztott lista, amik megadják, hogy ha egy modify event igazra teljesül, akkor a megadott objecteket milyen állapotba helyezze.

A szintaxis:  $\langle object_1ID \rangle : \langle state \rangle - \langle object_2ID \rangle : \langle state \rangle \dots$

**modify text** "/"-l elválasztott szövegek listája. Azon indexű szöveget foglalja kiírni a játék, ahányadik modify event teljesült.

FONTOS! Nem lehet egy-egy room és object aminek ugyan az az ID-ja, tehát a fájlokon keresztül is egyedi ID-kat kell alkotni.

## 1.2 Különböző event típusok

A következő eventeket értelmezi magától a program:

**go to**  $\langle room \rangle$  / **enter**  $\langle room \rangle$  Egyik szobából a másikba lép át.

**pick up**  $\langle object \rangle$  Ha egy object mozdítható, akkor áthelyezi a játékos eszköztárába.

**use**  $\langle object \rangle$  Megpróbál használni egy tárgyat. Hogy ennek a tárgynak a használata vált-e ki valamilyen eseményt azt ismételtelen az objects.stry-ben tudjuk meghatározni. Ezt az objects.stry-ben a következőképpen lehet definiálni: use

**use**  $\langle object1 \rangle$  **on**  $\langle object2 \rangle$  Megpróbál használni egy tárgyat egy másik tárgy kombinációjában. A pontos kombináció és a pontos történések az  $\langle object2 \rangle$ -ben kell, hogy definiálva legyenek. Ezt az objects.stry-ben úgy lehet definiálni, hogy azon az objecten hozzuk létre az eventet, amelyik az elszenvedője az eseménynek (tehát a nem az inventoryban lévő objecten), és a következő eventet helyezzük rá: use  $\langle objectID \rangle$

**look around/look at**  $\langle object \rangle$  Segítséget adhat a játékosnak, hogy most mit is kéne csinálnia, vagy hogy egy objectet hogyan tudna használni.

**Egyedi** Ha bármilyen más stringet írunk az eventhez, akkor fog az igazra teljesülni, hogyha a játékos pontosan azt a szöveget írja be.

A játékos mindig a jelenlegi szobában lévő tárgyakat és a szomszédos szobákat érheti el. Ahhoz hogy a szomszédos szobákban lévő tárgyakkal tudjon interakálni ahhoz már át kell lépnie a szomszédos szobába.