POSSIBLE

Tap Band Játéklogika megtervezése

Kornél Mészáros, Game Developer

BUDAPEST, HUNGARY / 12/19/2015

Agenda

Agenda

Proto részei

Játékelemek szétszedése MVC részekre
// Model: Data, State
// Controller
// View: GFX, UI (+ Effects, Audio)

Proto részei mechanikailag

Proto kötelező részei (legalább alapszinten)
// Tour (Concert (Song x N) x M)
// Tap
// Currency (Fans, cash, CPS = Coin per second)
// Equipment (tapre hat)
// Merch booth (cps) legalább aktív szinten

Csak a teljes játékba kell (protoba még nem)
// Booster, Token, IAP, Customization, Spotlight, Achievements, Audio, Settings stb.

Proto részei grafikailag

Proto kötelező részei

// Proto mechanikai elemek reprezentálása legalább placeholder jelleggel

- GFX: Zenészek, színpad, közönség helye akár cube/capsule-al
- UI: a mechanikák adatainak kijelzése:
- Protoban még minden játékállapot-információt mutatunk a UI-on!
  - Pl. időben hogy áll egy szám

Csak a teljes játékba kell (protoba még nem)

// Többféle zenész, közönség GFX-ek, animációk

// Particle Systems

// UI effektek, animációk

Végtelen játék módszerek

Képletekkel

+ könnyen módosítható
+ átlátható
+ kis memóriaigény

- nehezen ellenőrizhető egy adott eset
- script editor kell a gamedesignerek számára
- mindig “ugyanúgy működik”

Végtelen játék módszerek

Kegyetlen nagy XLS táblákkal

+ könnyen nyomonkövethető
+ egyszerű beolvasni és navigálni benne
+ párhuzamosan szerkeszthető
+ on-the-fly könnyebb bővíteni
+ a játék előrehaladtával változhatnak az elvek

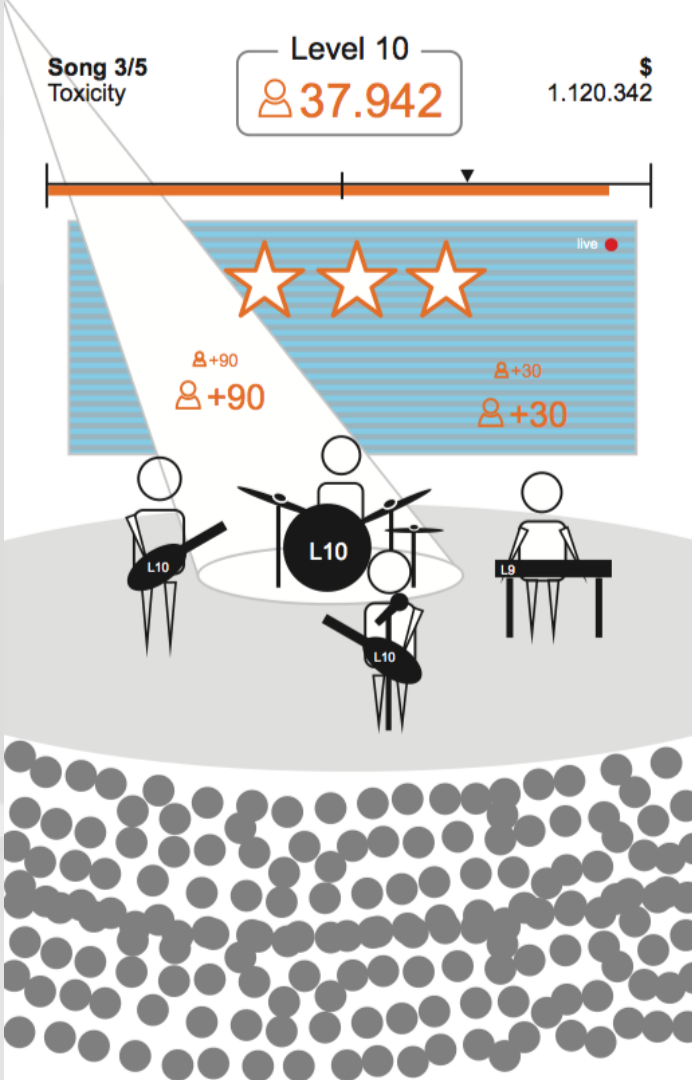
- véges: idővel bővíteni kell (“lejjebb húzzuk az xls sorokat”)
- nagy adathalmaz - memóriaigényt kezelni kell

Proto játékelemek szétszedése MVC szerint

Song

Beszédesebb a Song elnevezés, mint a Level vagy Szakasz
~ Egy Tap Titans monster

SongData
// Id
// Title
// Goal Fan
// Duration (állandó?)
// Coin Reward
// IsEncore (~Boss battle)
› A song tárolja magáról, vagy a SongController/ConcertController szabályozza?



Proto játékelemek szétszedése MVC szerint

Song

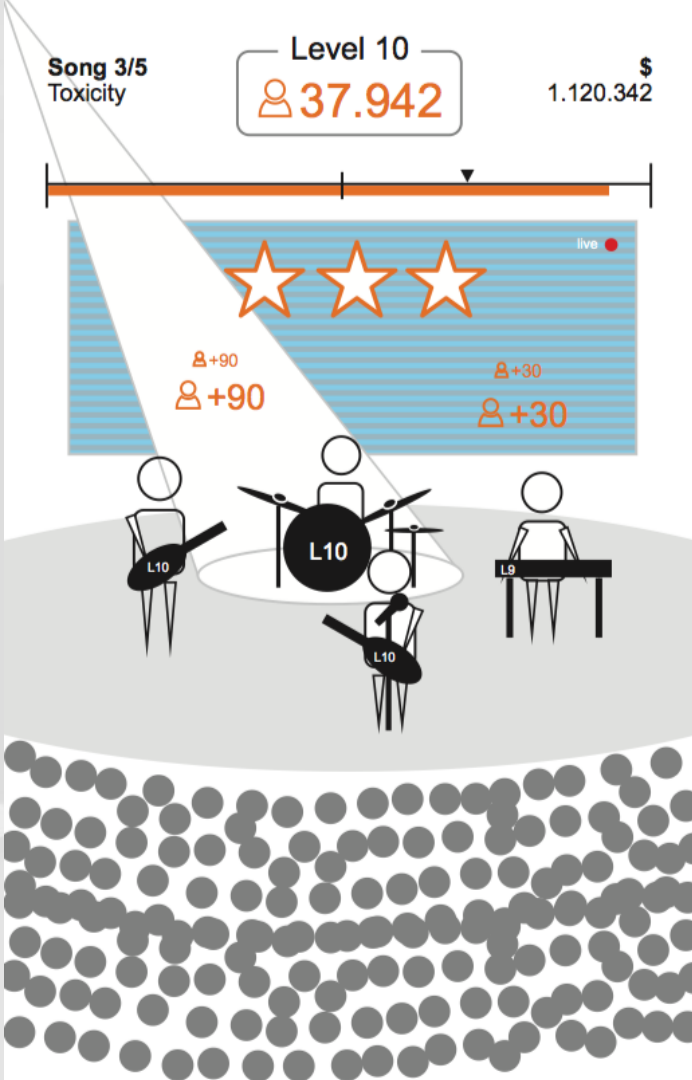
SongState

// Current Song Id (mentjük?)

// Lekérdezések az aktuális song-hoz

// Szám betöltése Id alapján

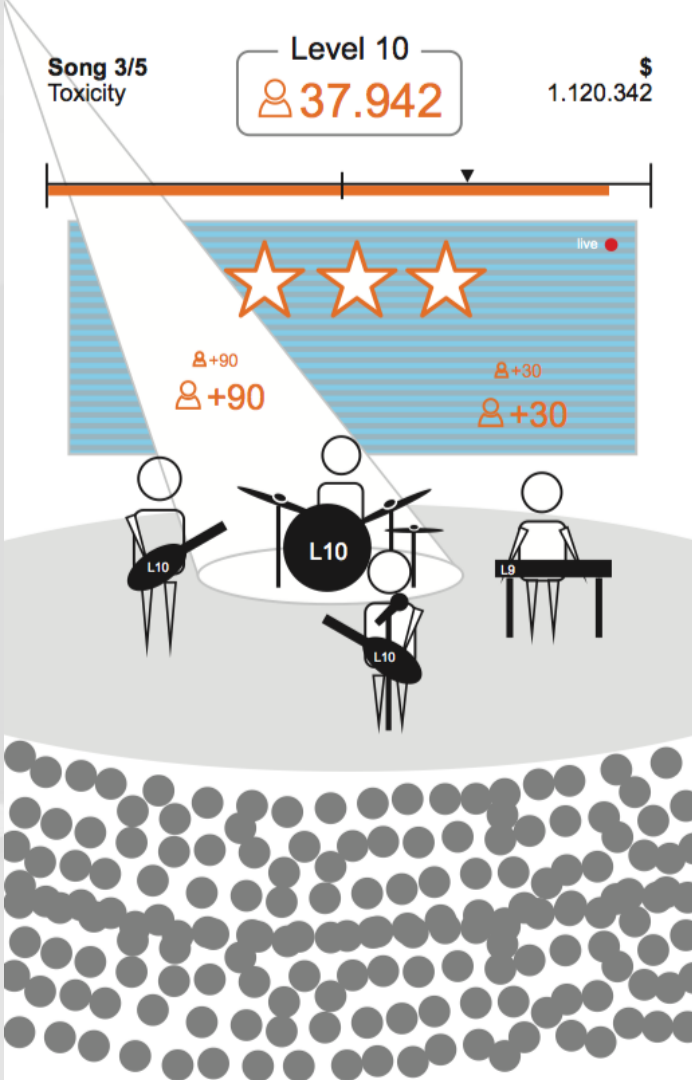
// Információ biztosítása a jelenlegi és a következő számról



Proto játékelemek szétszedése MVC szerint

Song

SongController
// Szám indítása, idő követése
// Aki a számot indítja feliratkozik
// Visszaszólás a feliratkozónak, akkor ha
› vége a számnak (lejárt az idő)
› goal fan elérve
// Jutalom realizálása



Proto játékelemek szétszedése MVC szerint

Song

SongGFX

// Van egyáltalán?

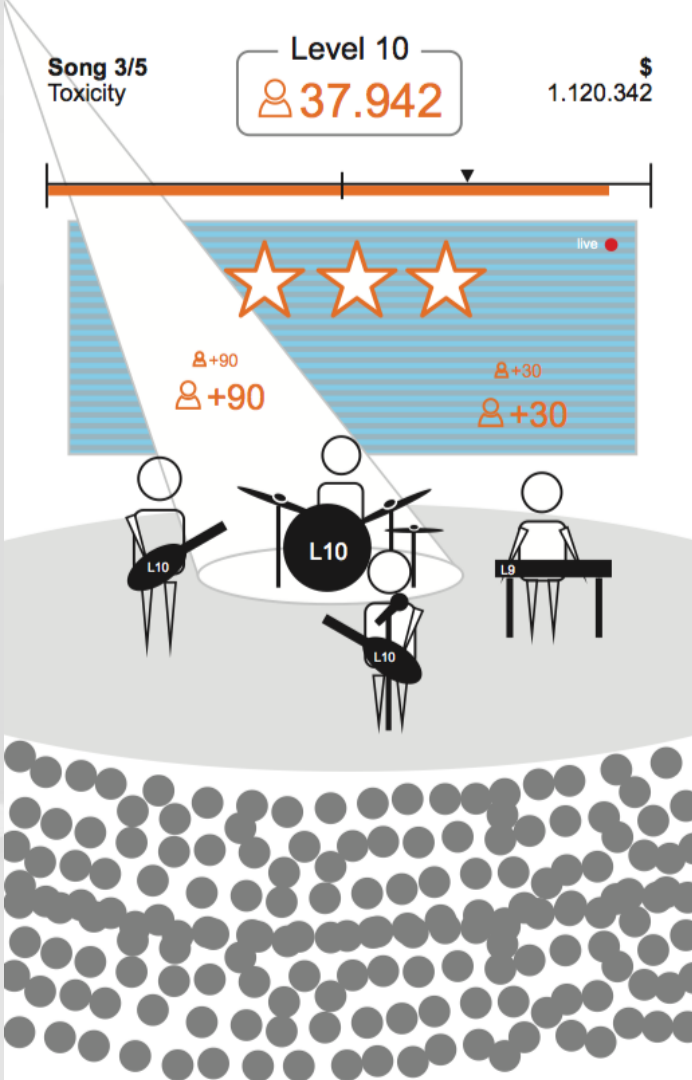
// Fényekkel?

SongUI

// Id, title, goal fan, current fan, duration, current time, isEncore?

// Lehetőleg progress barokkal

// Title random generált?



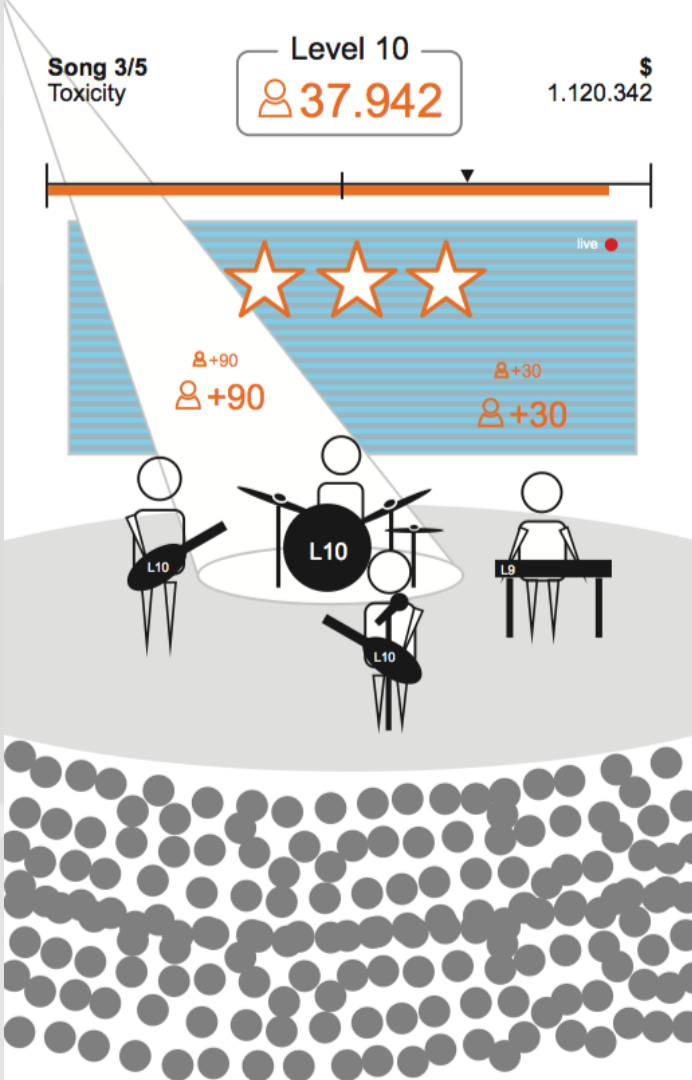
Proto játékelemek szétszedése MVC szerint

Song

Proto után

// Star Threshold
- › fan szintek szerint és ennek kiértékelése

// Idő tartalék
- › Idő előtti teljesítéskor
- › visszamaradó idő Boosterbe?

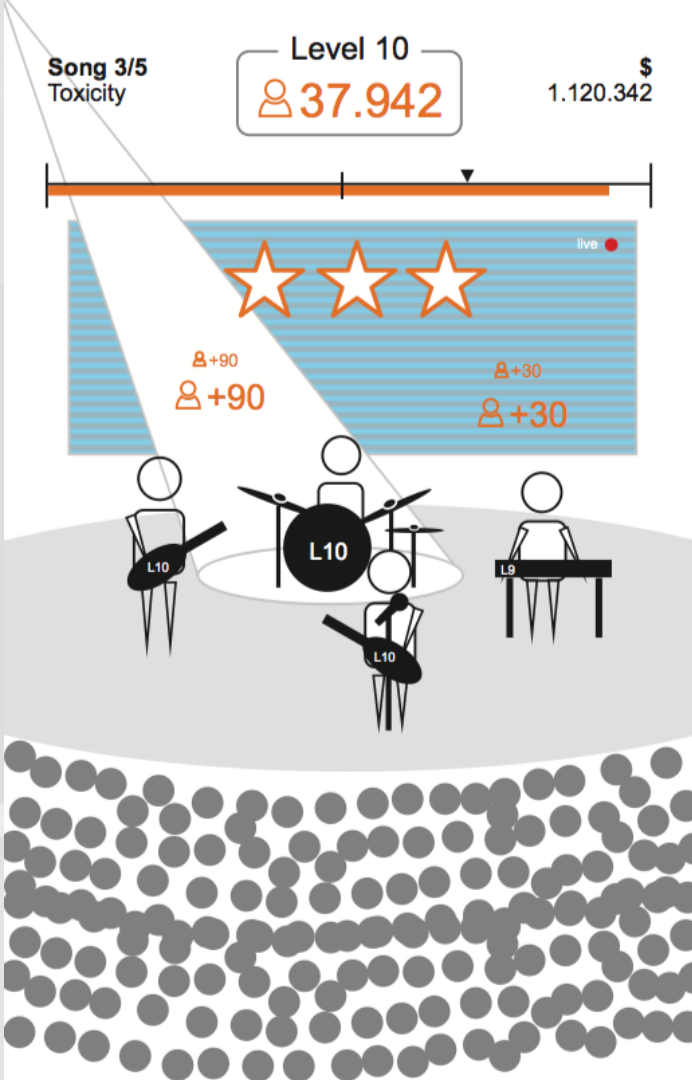


Proto játékelemek szétszedése MVC szerint

Concert

Show, Játékmenet szó nem szerencsés, ezért Concert
~Tap Titans N szörny-sorozat a végén a boss-szal

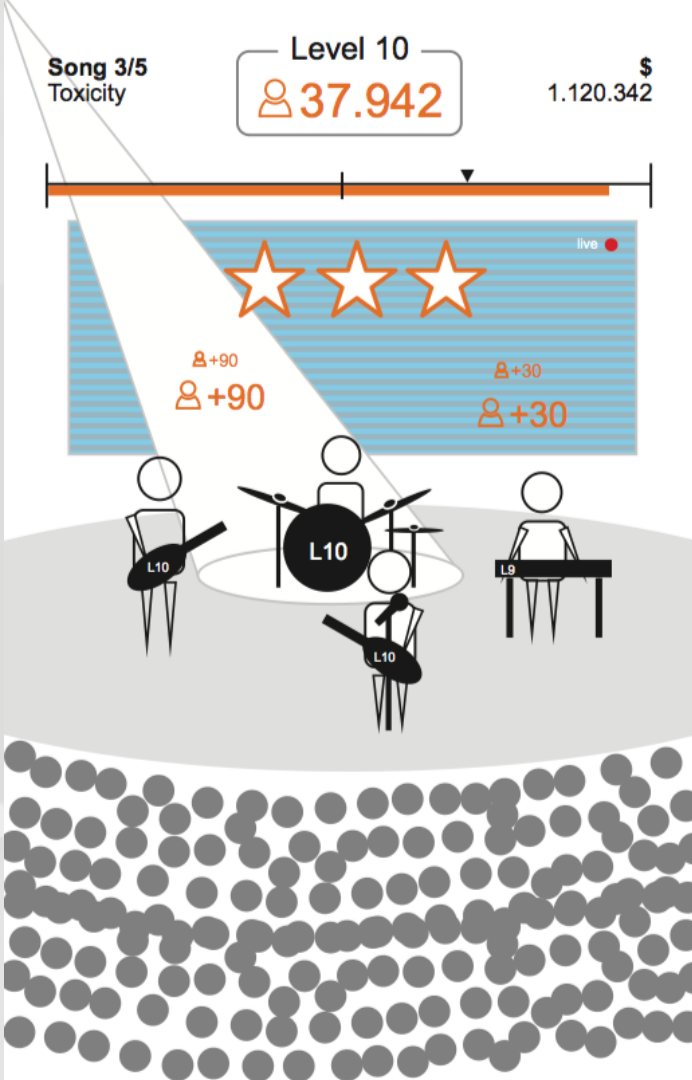
ConcertData
// Id
// Name
// SongIds [N] (Setlist)
// Fan Reward
// Coin Reward



Proto játékelemek szétszedése MVC szerint

Concert

ConcertState
// Current Concert Id
// Last Completed Song Id (itt mentünk song állapotot)
// Current Song (objektum, nem Id!)
// Információk biztosítása az aktuális Concert-ről



Proto játékelemek szétszedése MVC szerint

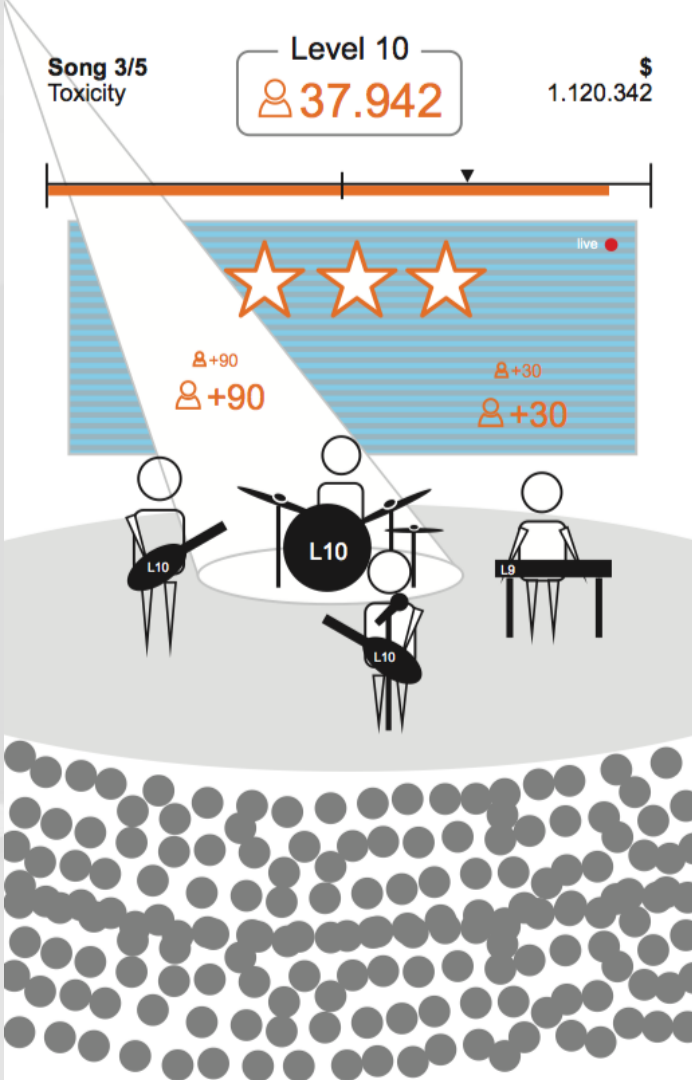
Concert

ConcertController

// Kapocs a számok és a turné között

// Számok menedzselése (új szám indítása, aktuális szám állapotának figyelése, szám pause nézetváltáskor)

// Folyamatosan menjenek a számok (ha valamit manuálisan kell indítani, akkor kilépési pontot hagyunk a játékosnak)



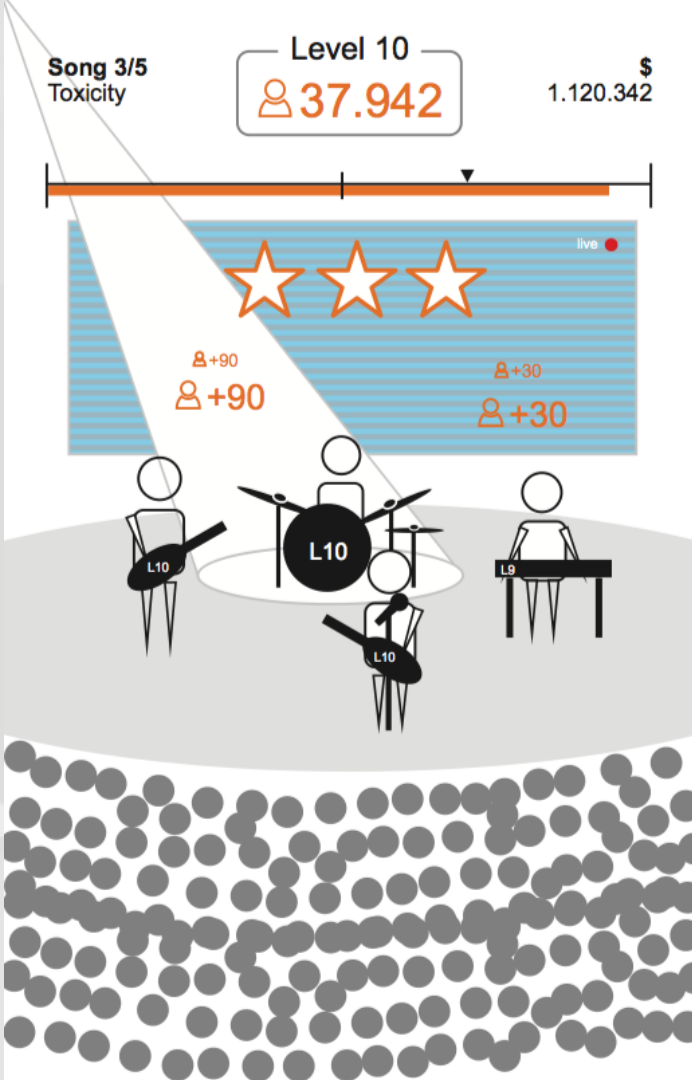
Proto játékelemek szétszedése MVC szerint

Concert

ConcertGFX
// Később

ConcertUI
// Song progress (current / max)
// Esetleg reward

Proto után
// Változó ConcertGFX/StageGFX
Stageld attribútum alapján, X szintenként



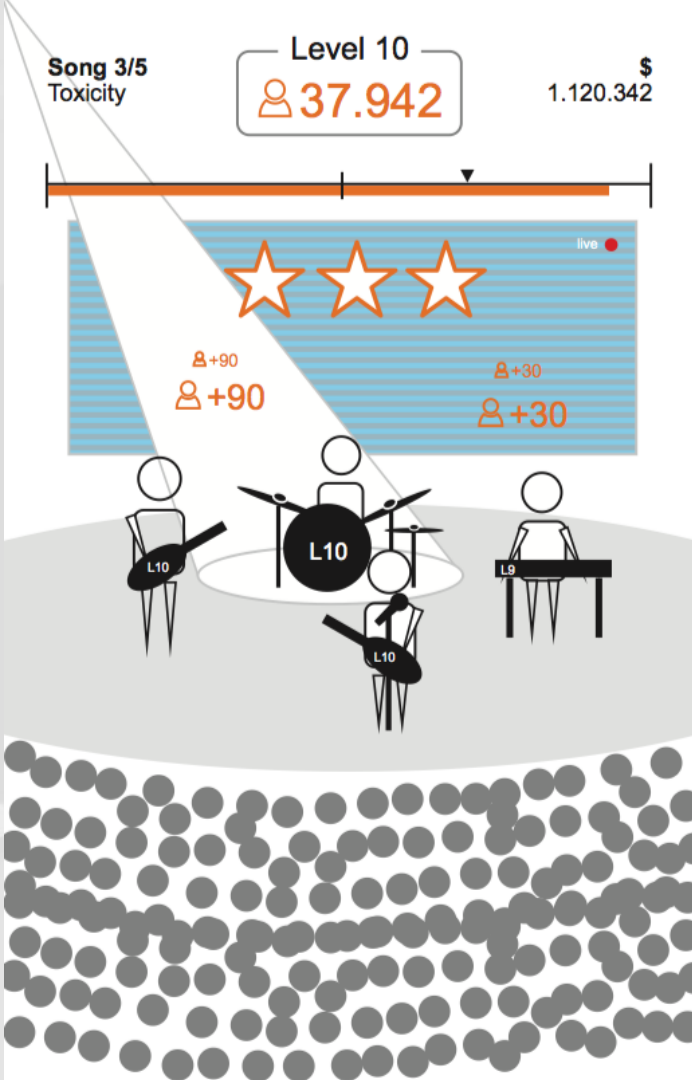
Proto játékelemek szétszedése MVC szerint

Tour

Prestige szint megfelelője

TourData
// Kell külön TourData?
// Elég lehet csak a jutalom szorzó a TourState alapján

TourState
// CurrentTourIndex
// Származtatott Tour bonus értékek



Proto játékelemek szétszedése MVC szerint

Tour

TourController

// Annak vizsgálata, hogy lehet-e már következő turnét indítani (prestige szintet lépni)

// Szintlépés biztosítása

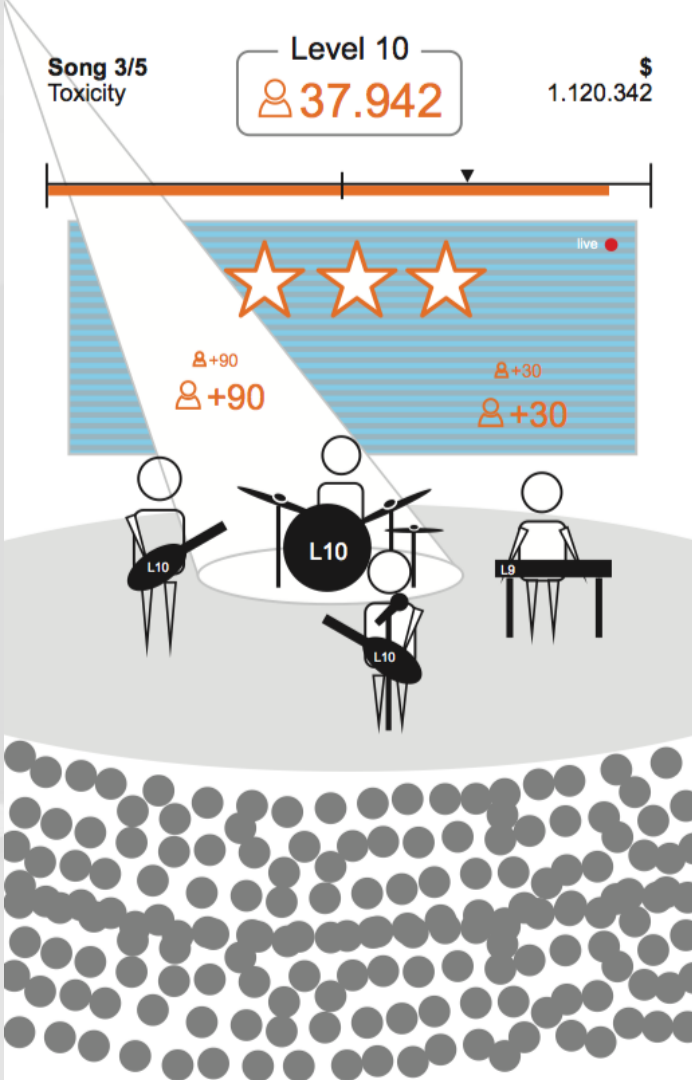
// Bizonyos currency továbbvitele

TourGFX

// később

TourUI

// Hanyadik turnénál járunk



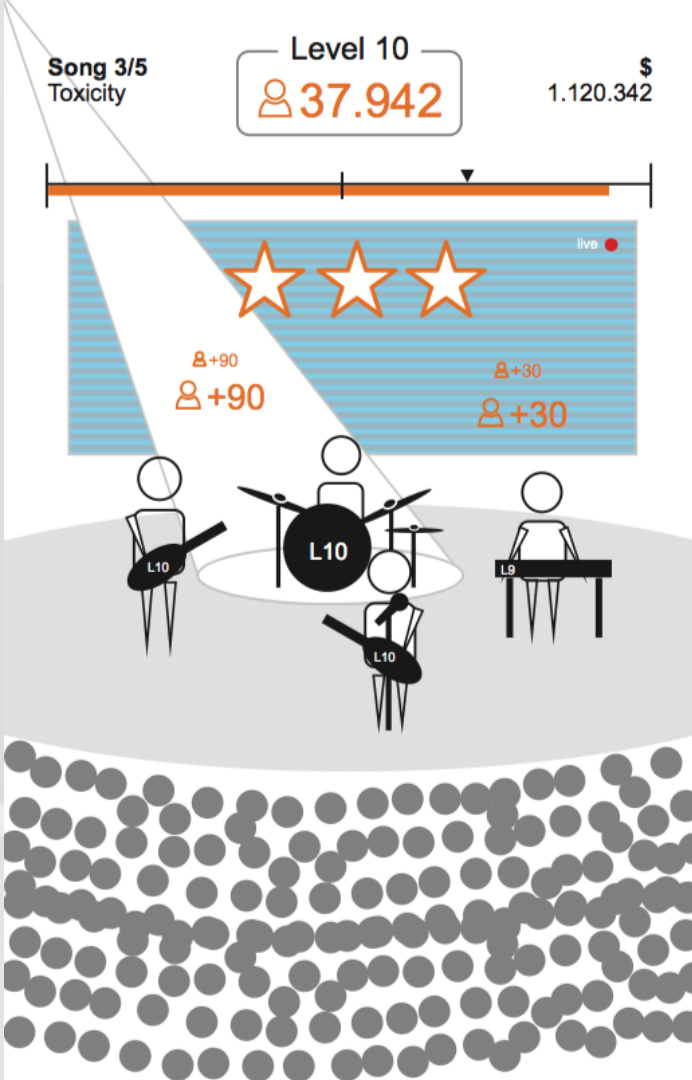
Proto játékelemek szétszedése MVC szerint

Tour

Proto után, opcionálisan

// Tour sorszám alapján egyre sötétedő napszak (TourGFX/ Environment)

// Egy idő után már éjjel mehetnének a koncertek

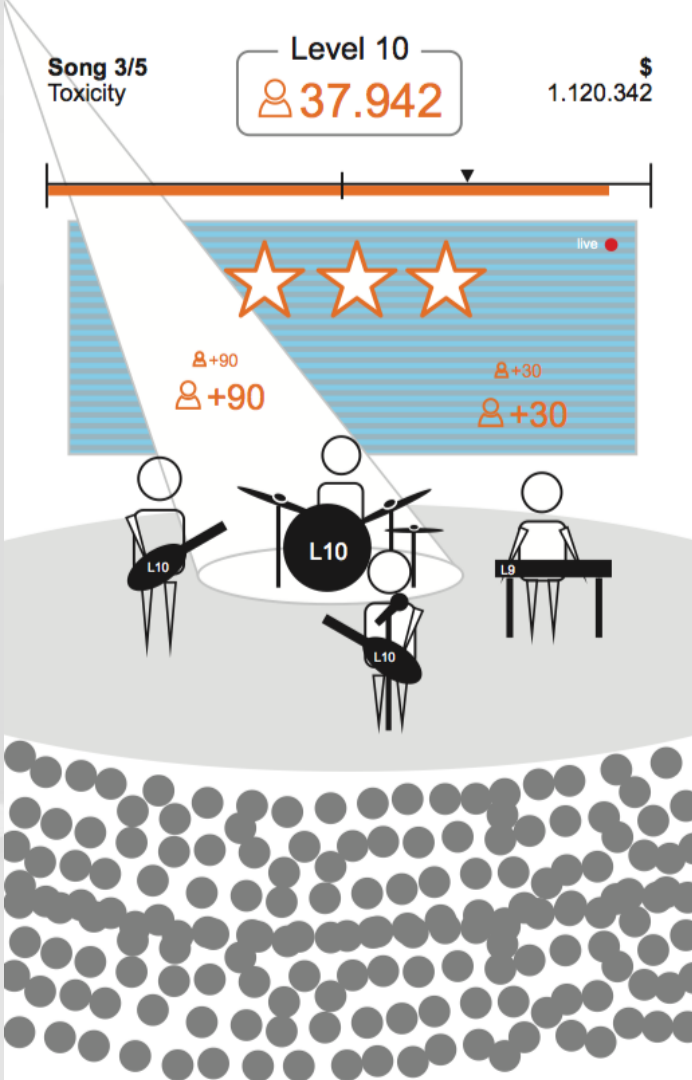


Proto játékelemek szétszedése MVC szerint

Tap

TapData
// FanForTapBase, ha kell
// Inkább legyen 1 az alap érték és nem kell

TapState
// Nincs mit menteni, mert származtatott érték
// Információk biztosítása a jelenlegi Tap értékről
- Equipmentek és
- Boosterek alapján

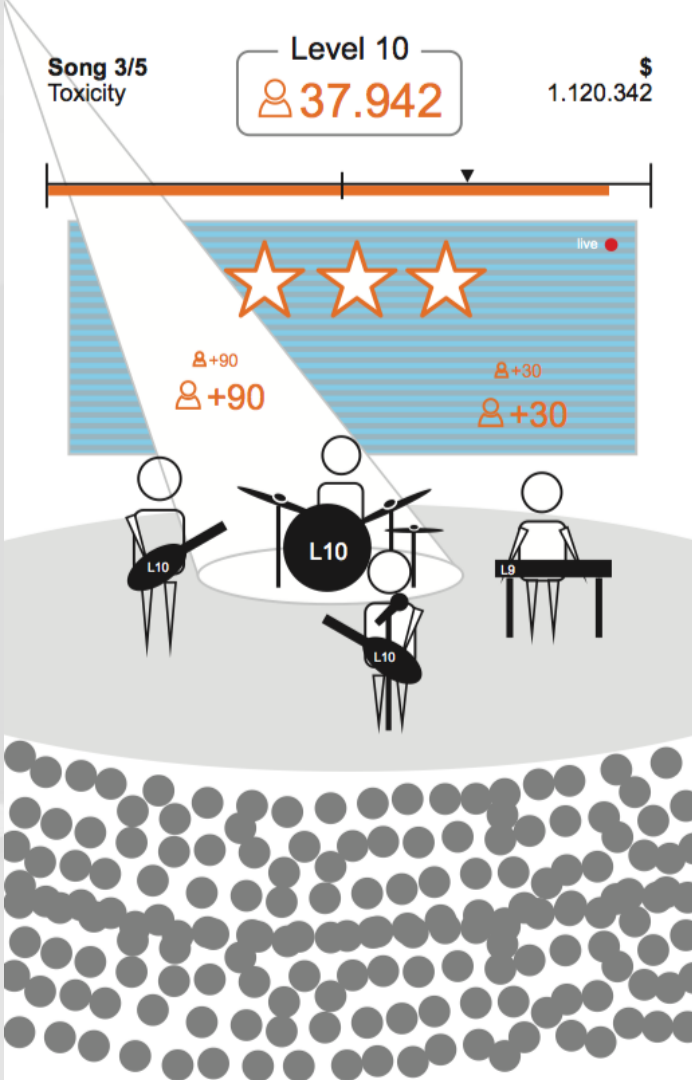


Proto játékelemek szétszedése MVC szerint

Tap

TapController
// Tap érzékelése
// Tapért jutalom (FanForTap) számítása
// A jutalom odaadása
› a SongControllernek és
› a CurrencyControllernek

TapGFX
// később



Proto játékelemek szétszedése MVC szerint

Tap

TapUI

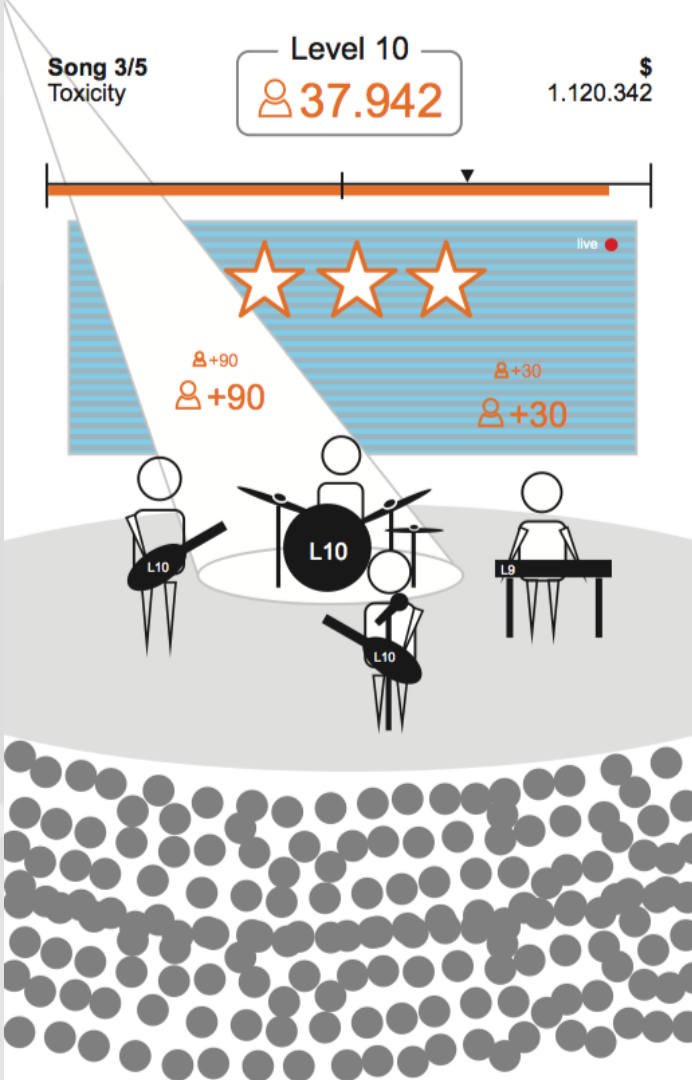
// Tapért kapott fan jutalom kijelzése szövegesen

Tap(UI)Effects

// Már a protoban is fontos
a felhasználói interakcióra adott feedback

// Tap effekt

- **pl. szikra, villanás**
- **lehet PS vagy UI effekt**



Proto játékelemek szétszedése MVC szerint

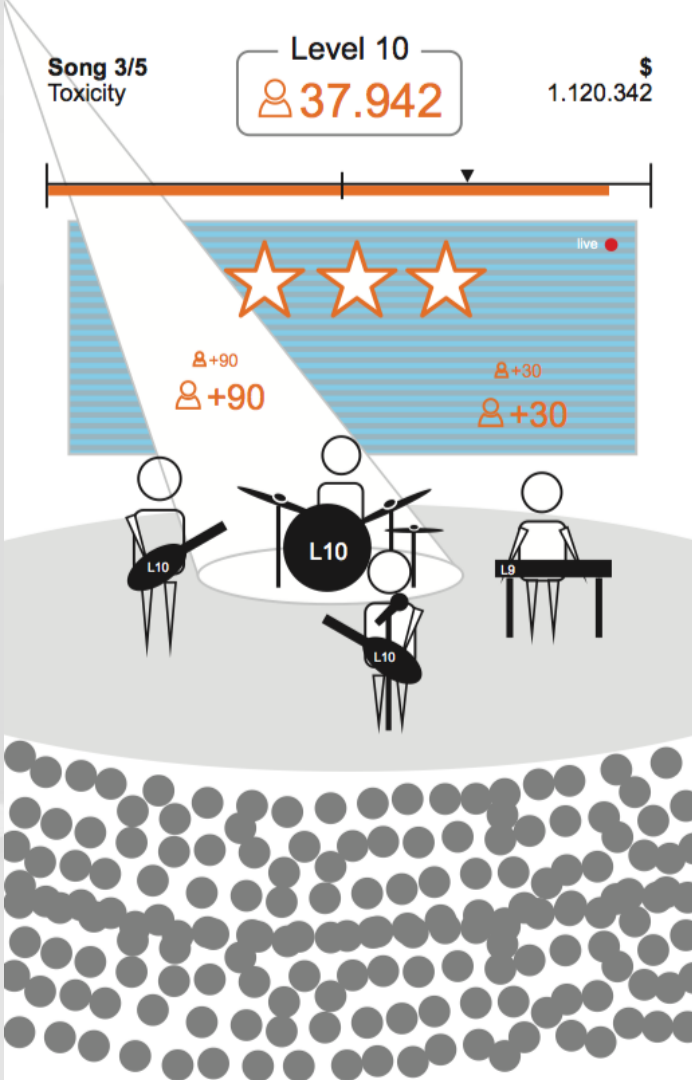
Tap

Proto után

// Multitouch kezelés

// TapGFX + Effects

- › Karakterek másfajta animmal?
- › Közönségben valaki felugrik / tapsol?
- › Karakterek megvilágítása alulról / alsó reflektorfények villogtatása?



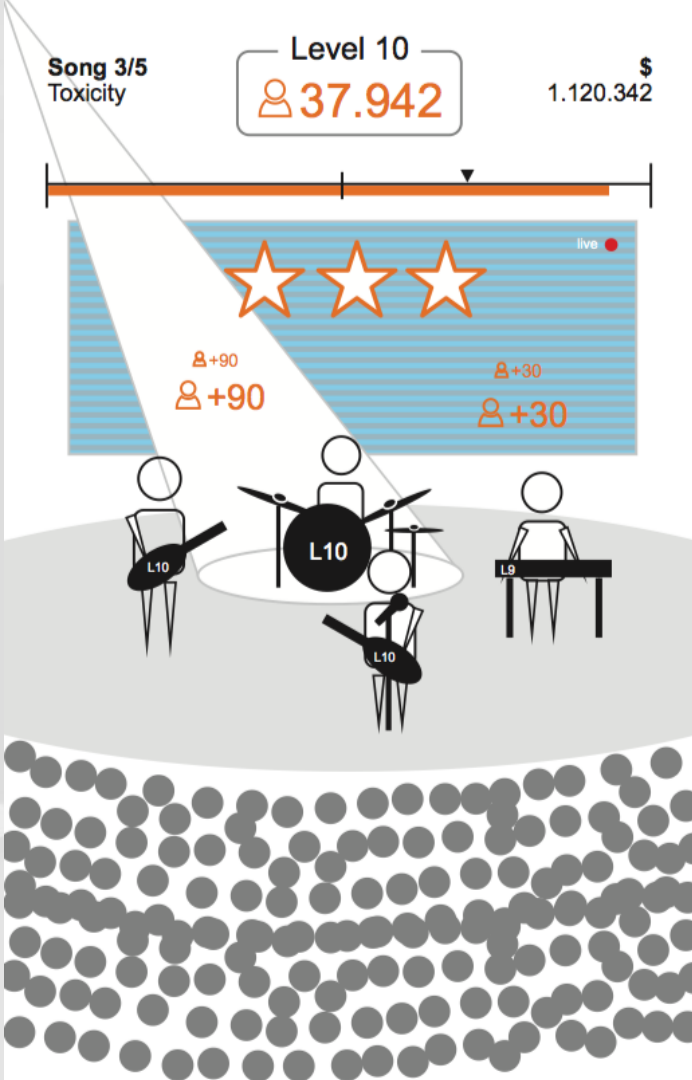
Proto játékelemek szétszedése MVC szerint

Currency

CurrencyData
// Nincs mit tárolni

CurrencyState
// Fan
// Coin
// (Token)

// Transient values!



Proto játékelemek szétszedése MVC szerint

Currency

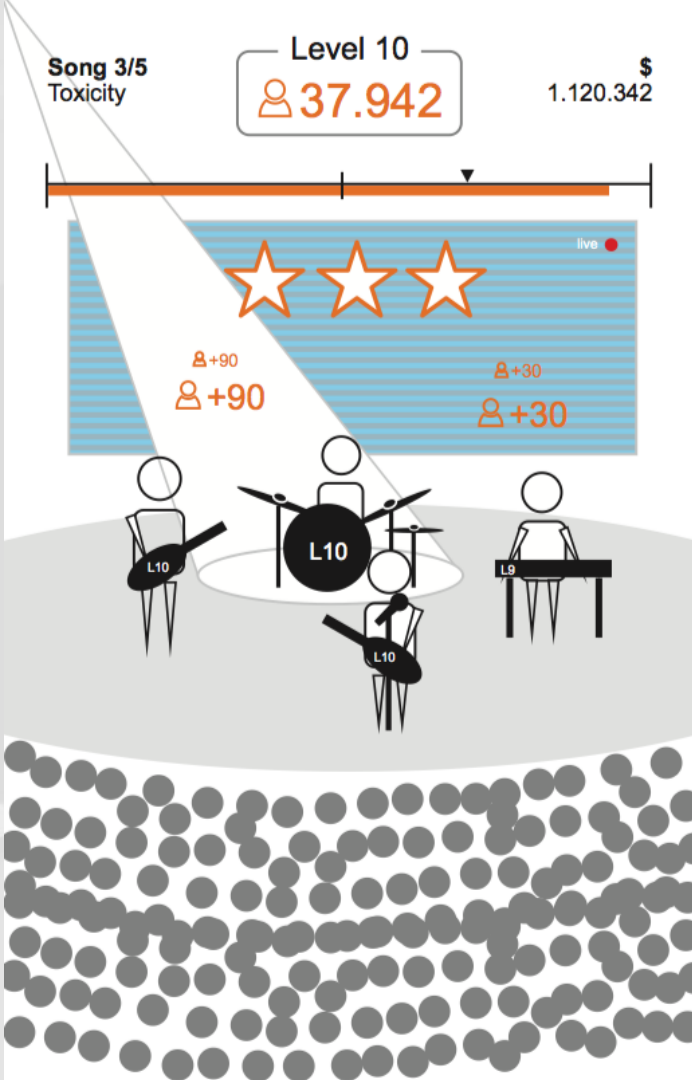
CurrencyController

// Lekérdezések,
pl. egy bizonyos összegű vásárlás elvégezhető-e?

// State értékek módosítása

// Jelzés a módosításról
(értesítünk mindenkit aki az aktuális mennyiségtől függ)

// CPS számolása a Merch Booth alapján és jóváírás



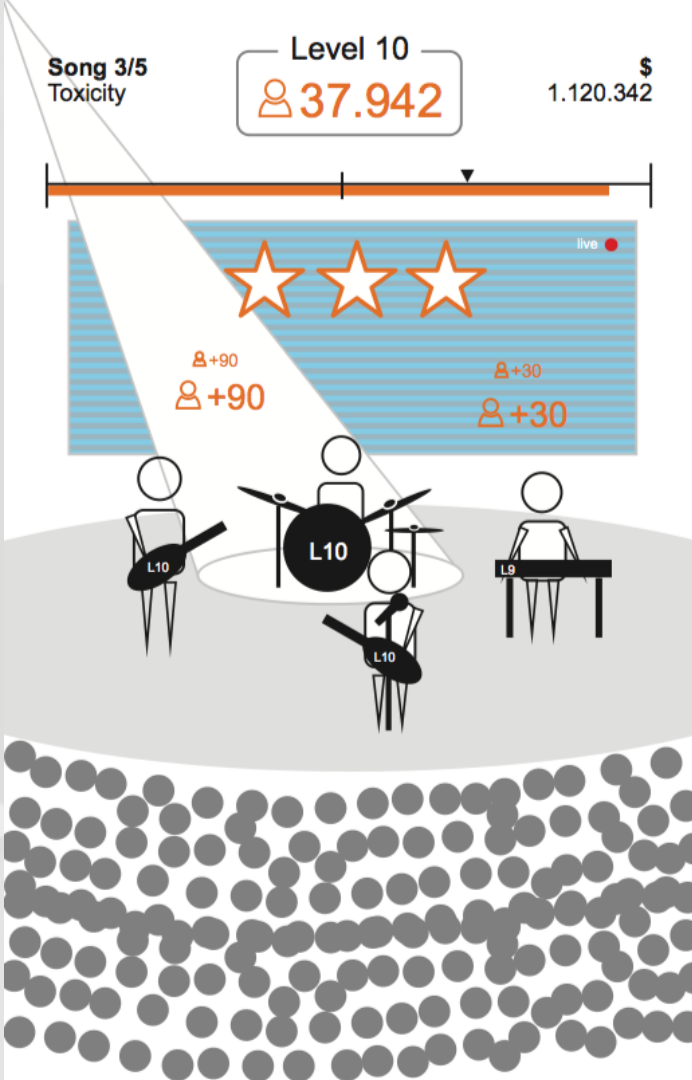
Proto játékelemek szétszedése MVC szerint

Currency

CurrencyGFX
// később

CurrencyUI
// State értékek kijelzése
› Fan, coin, token

Currency(UI)Effects
// State értékek módosulásának jelzése a UI-on, pl. scale anim



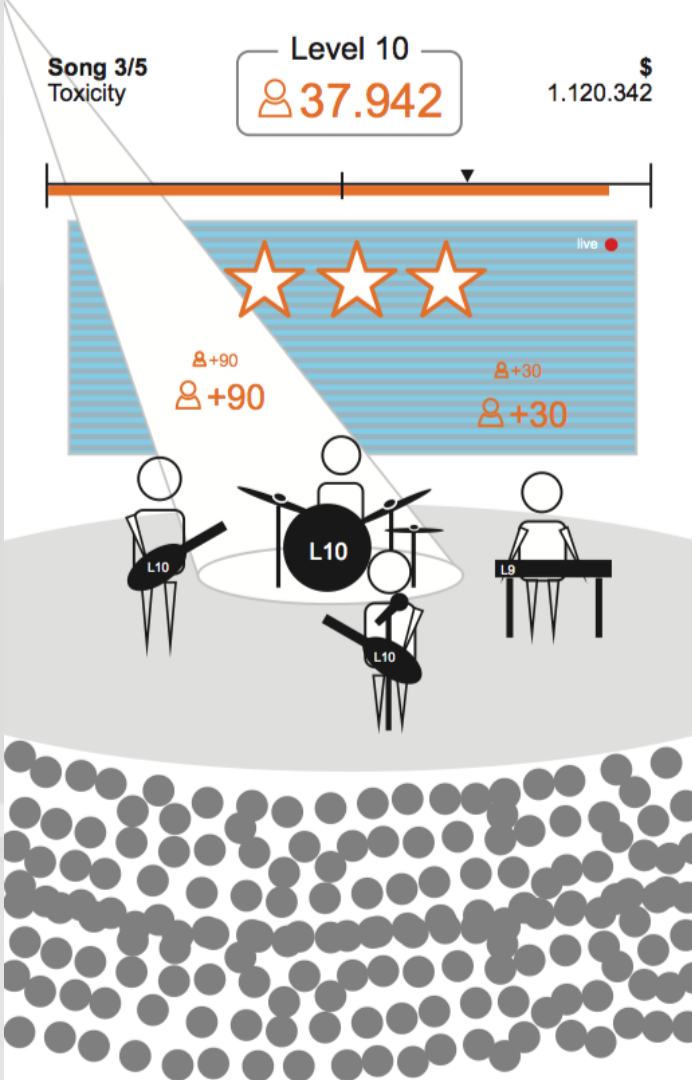
Proto játékelemek szétszedése MVC szerint

Currency

Proto után

// Összekötés a közönséggel:
Fanok mennyisége alapján a közönség méretének szabályozása - CurrencyGFX

// Harmadik currency:
Token → IAP

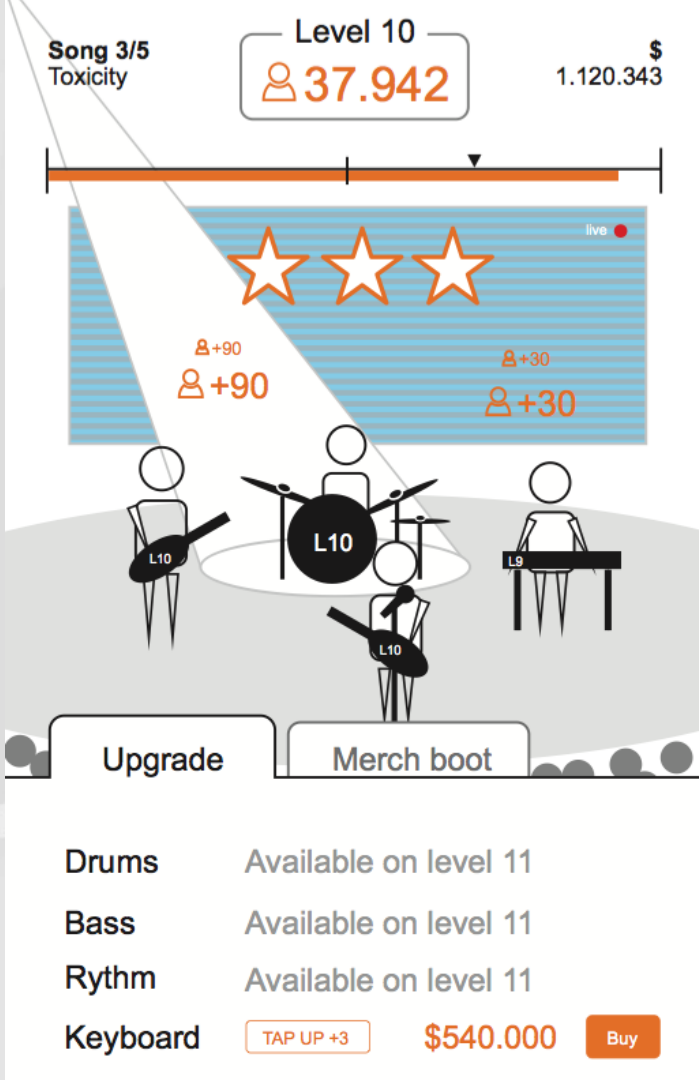


Proto játékelemek szétszedése MVC szerint

Equipment

Tapért kapott jutalom mennyiségét befolyásolja

EquipmentData
// Category - lehet enum
// Category Name (hangszer kategória pl. Solo Guitar)
// LevelData
- Id
- Name? (hangszer neve
  pl. Michael Amott Tyrant Blood Storm Electric Guitar)
- FanRequirement (efelett válik elérhetővé)
- UpgradeCost
- FanValue (ennyit ér a FanForTap-ba)



Proto játékelemek szétszedése MVC szerint

Equipment

EquipmentState

// Minden kategóriához tároljuk, hogy melyik a legmagasabb unlockolt hangszer

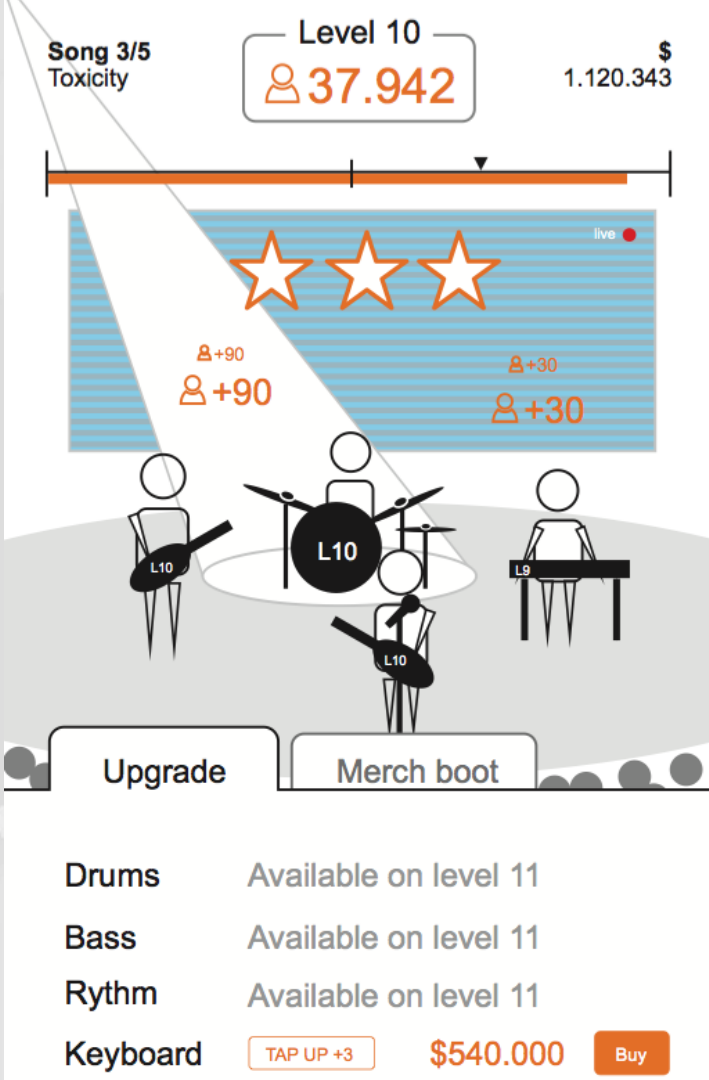
// CategoryId + EquipmentId lista

// Lekérdező függvények a jelenlegi equipment állapothoz

EquipmentController

// Equipment upgrade-k vásárlását teszi lehetővé

// Equipment változásról értesít, pl. a TapController-t



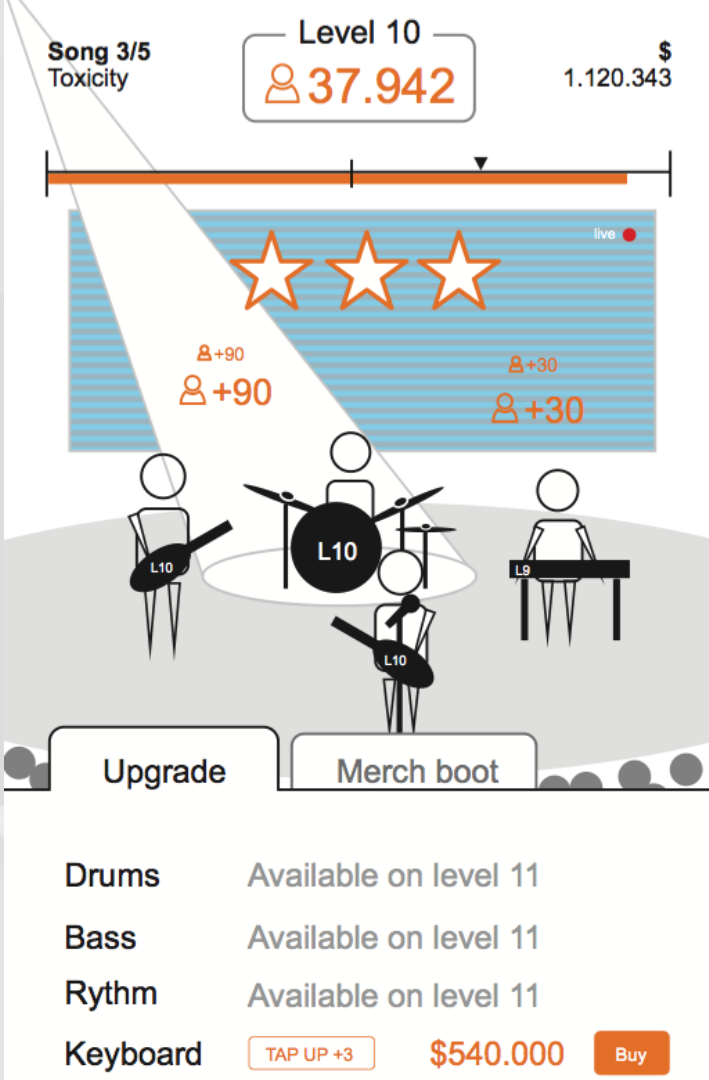
Proto játékelemek szétszedése MVC szerint

Equipment

EquipmentGFX
// Később

EquipmentUI
// UI ablak az equipmentekről
// Típus szerint sorokba, aktuális szint mutatása

Proto után
// Összekötése karakterekkel: ha változik a hangszerek grafikája EquipmentGFX vagy Character GFX



Proto játékelemek szétszedése MVC szerint

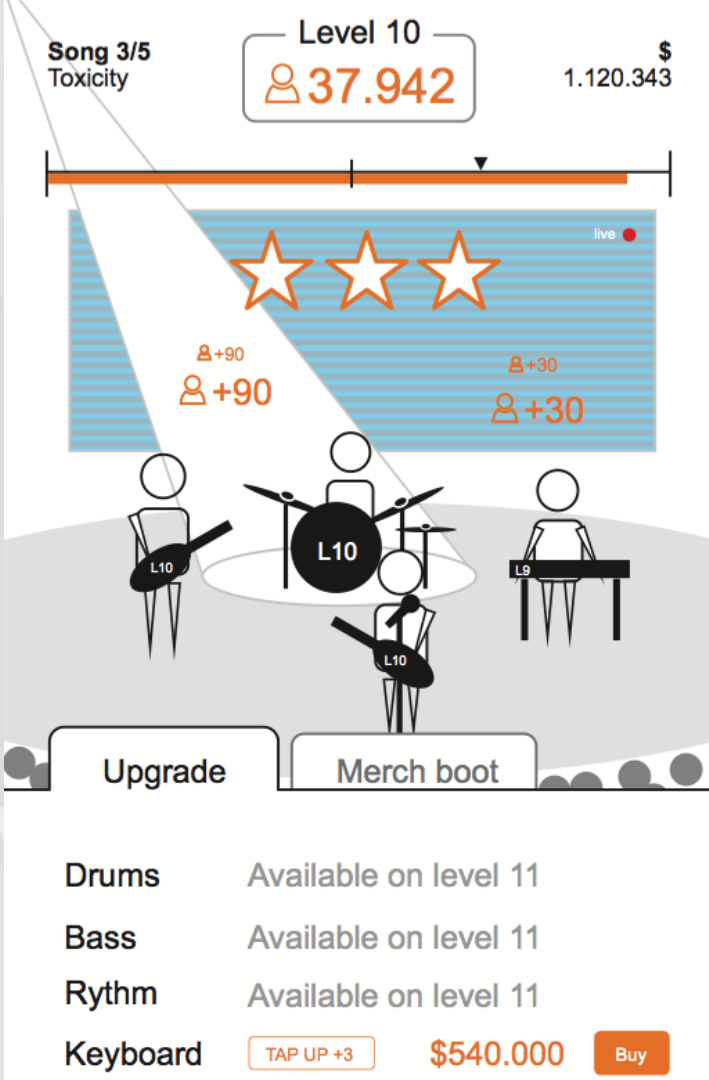
CPS: Coin Per Second

Amikor a játékos nem játszik a játékkal,
akkor is termelődik a pénze
Játékbeli reprezentáció: Merch Booth

Mechanikailag két részből áll:

// Merch Booth
- **Meg van nyitva az alkalmazás**
- **Játék futása során másodpercenként currency-t termel**
- **A CurrencyController végzi a CPS összegyűjtését**

// Offline Coin
- **Alkalmazás bezárva vagy háttérben fut (paused)**
- **Játékba visszatéréskor kap currency-t a játékos**



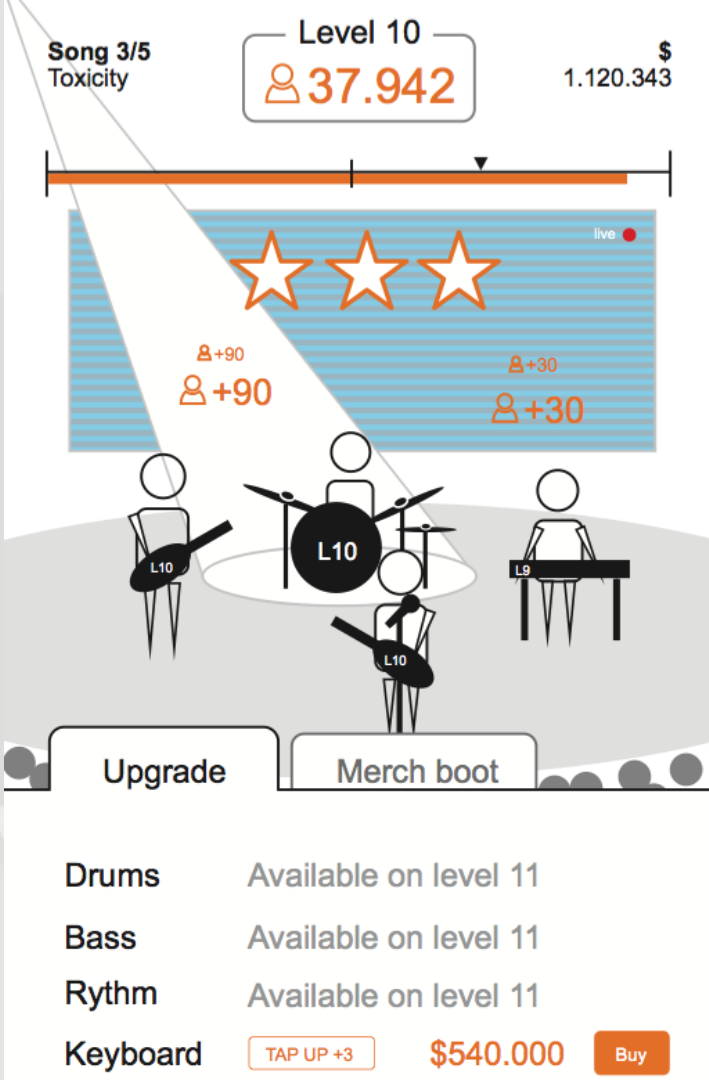
Proto játékelemek szétszedése MVC szerint

CPS: Merch Booth

Elnevezés
// Merch / MerchBooth / Merchandise

MerchData
// CPSData
- Level
- Coin Per Second
- Upgrade Cost

// CapacityData
- Level
- Capacity
- UpgradeCost

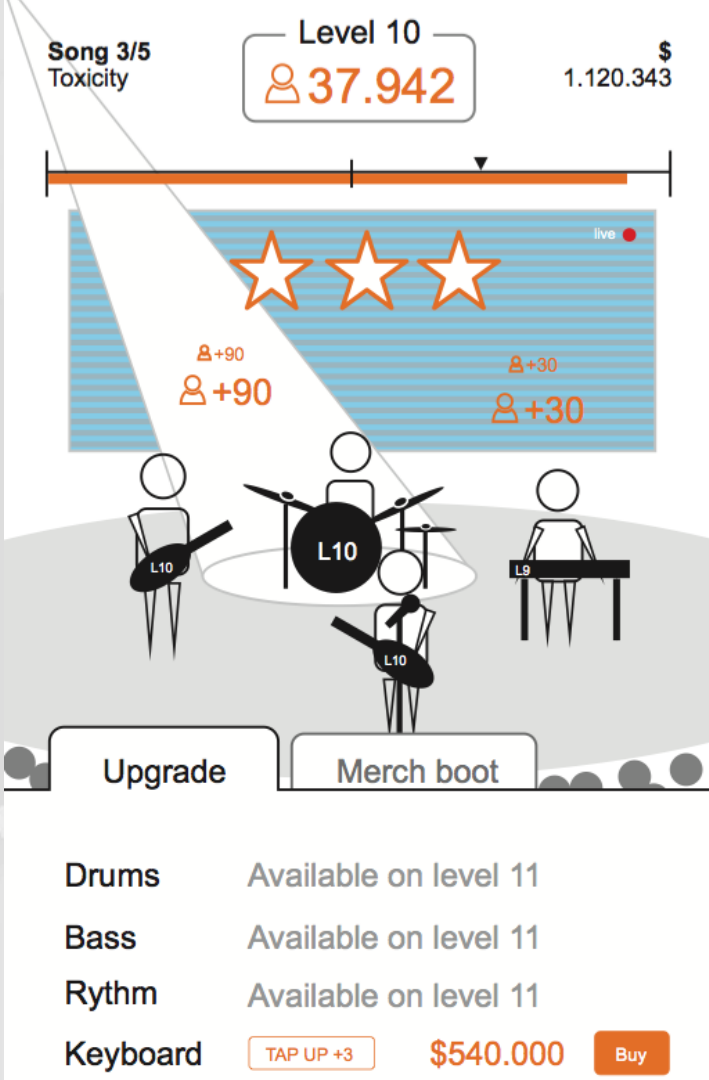


Proto játékelemek szétszedése MVC szerint

CPS: Merch Booth

MerchState
// Current CPS Level
// Current Capacity Level
// Lekérdező függvények

MerchController
// CPS lekérdezése, vizsgálat a Capacity-re stb.



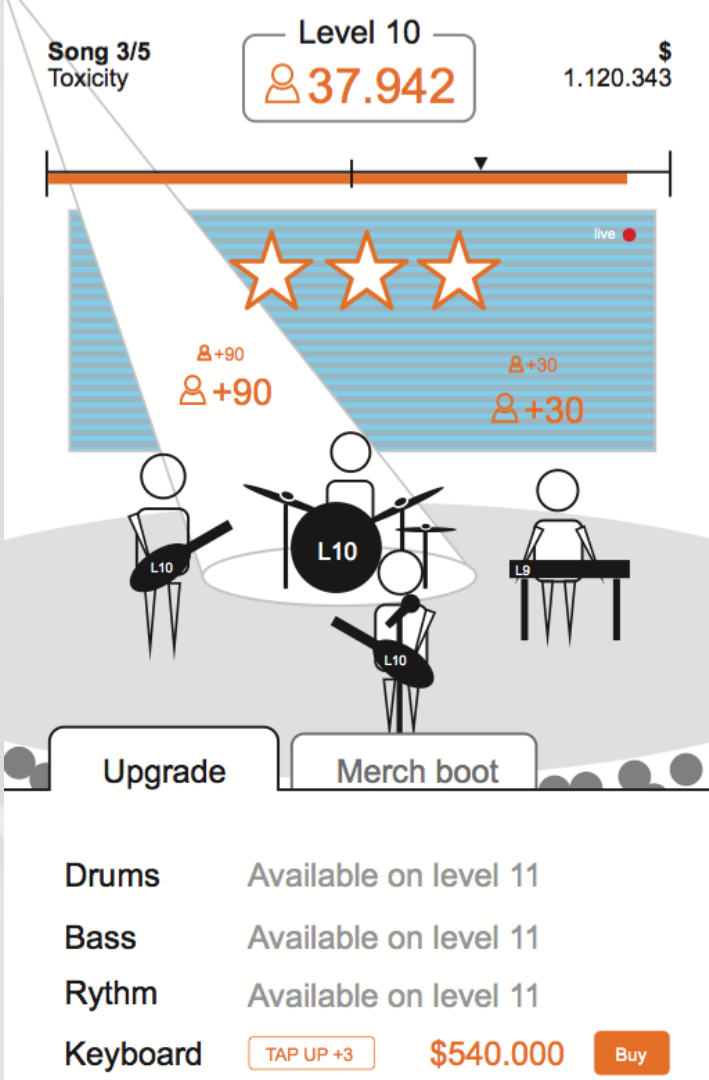
Proto játékelemek szétszedése MVC szerint

CPS: Merch Booth

MerchGFX
// később

MerchUI
// Equipment-hez hasonlóan egy ablak ahol
// Upgradelhető
› a CPS és
› a Capacity

Proto után
// Backstage/ próbaterem/ MerchGFX változása a MerchState progress alapján



Proto játékelemek szétszedése MVC szerint

CPS: Offline Coin

OfflineCoinState

// Nyilvántartjuk, hogy mennyit termelt eddig

// Ha nem gyűjti be, továbbviszi a következő sessionre

OfflineCoinController

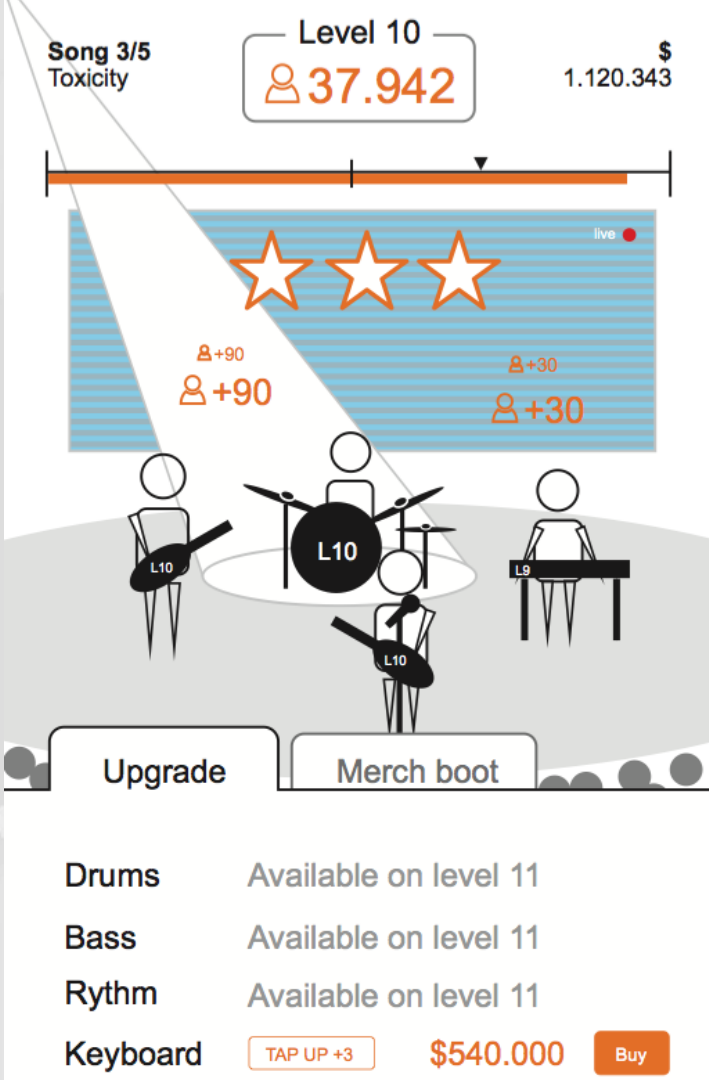
// Kilépéskor elmentjük a kilépés időpontját

// Pause-ba küldéskor is

// Játékba visszalépéskor kiszámoljuk a jutalmat az eltelt idő alapján

// Skálázódik a játék progress alapján

// Mi alapján számoljuk?



Proto játékelemek szétszedése MVC szerint

CPS: Offline Coin

OfflineCoinGFX

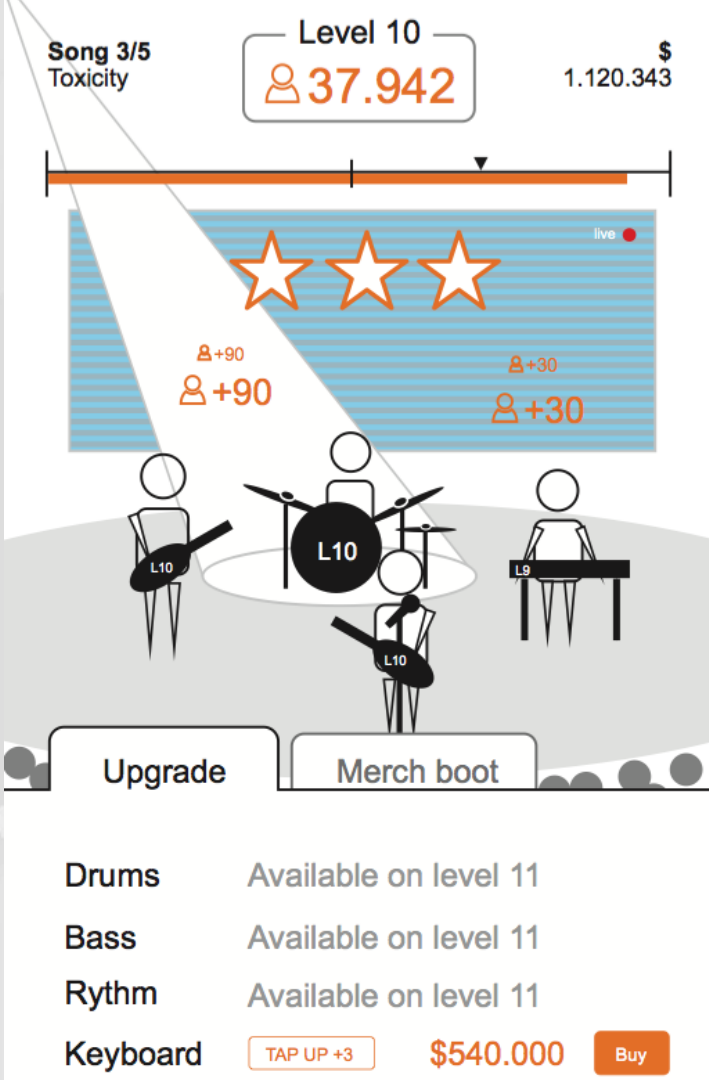
// nincs

OfflineCoinUI

// Pl. Egy gomb, amit ha a játékos megnyom megkapja az offline jutalmat

Proto után

// Max Kapacitás nyilvántartása és fejleszthetősége

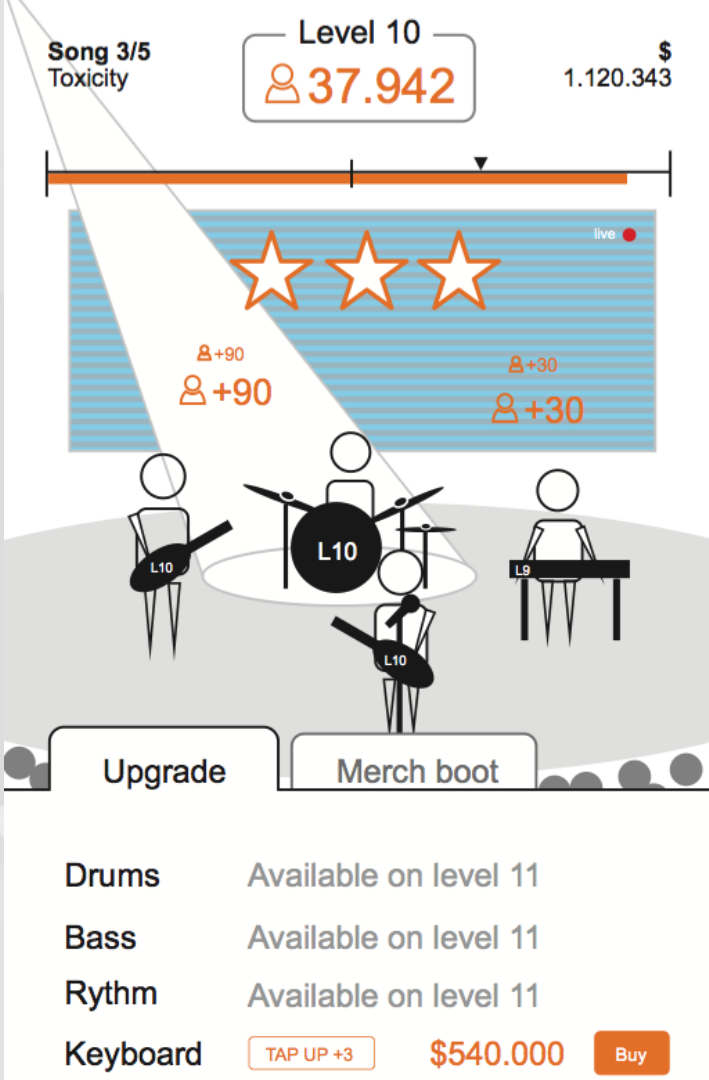


Proto játékelemek szétszedése MVC szerint

Debug

DebugController
// Kommunikál a megfelelő controllerekkel
// Coin, fan, level/upgrade adása gombra

DebugUI
// Debug funkciók kivezetve egy ablakra
// Ki-be kapcsolható egy gombbal



Proto játékelemek szétszedése MVC szerint

Speciális MVC elemek

GeneralData

// Globális adatok, melyek egyik nagyobb csoporthoz se tartoznak igazán

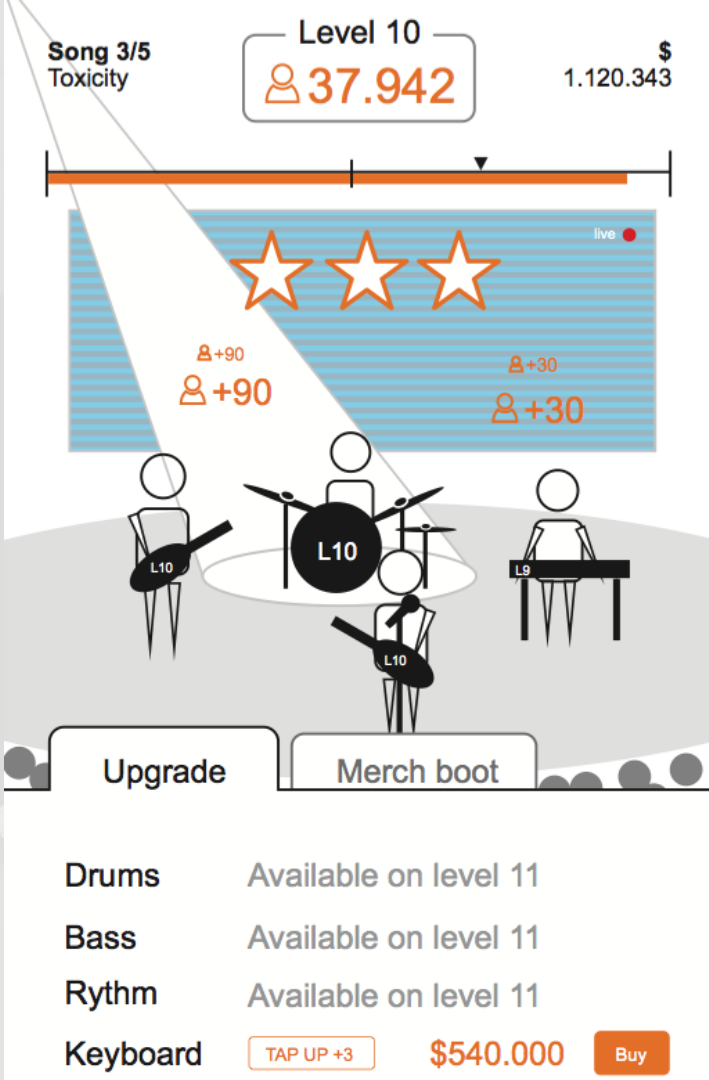
// Ne kódba égessünk be konstansokat, hanem vezessük ki őket, hogy gamedesigner is tudja módosítani

Game Controller

// Teljes játékra kiterjedő, összefogó és elosztó folyamatok

// Ha lehet kerüljük a használatát

// Pl. Nézetváltáshoz, idő kezeléshez



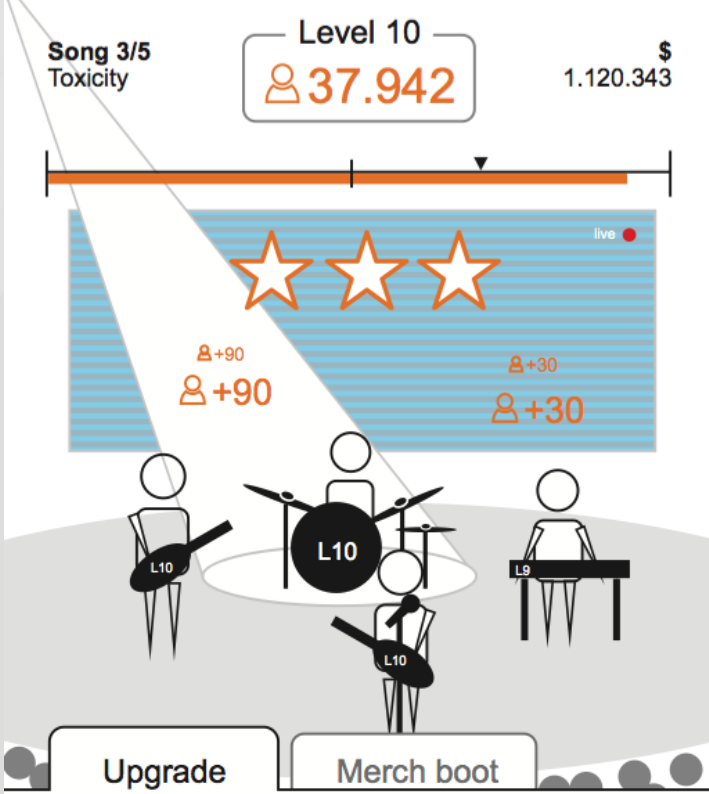
Proto játékelemek szétszedése MVC szerint

Proto után

Mechanikák
// Booster
// Spotlight

Grafikai elemek
// Színpad, színpad elemek
// Karakterek (customizations?)
// Közönség víz shaderrel

Egyéb
// Nézetváltás színpad és upgrade rész közt
// Audio
// Teljes UI bekötés



Drums	Available on level 11		
Bass	Available on level 11		
Rythm	Available on level 11		
Keyboard	TAP UP +3	$540.000	Buy

Extra

Közönséges víz shader

Közönség mozgatása víz shaderrel / fluid sim
// Vertexekre a 2D emberek
// Low poly
// Állítható sűrűség
// Állítható sebesség
  › Lassabb szám alatt lassabb hullámzás
  › Gyorsabb alatt ugrálás
// Random vertex felugrása hívásra

Színpad vízben?