

# Příklad reportů.

Tento report byl založen na reportech odevzdaných v rámci PHA 2014, původními autory jsou Vojta Micka a Martin Kopřiva. Upravil tým PHA.

Jedná se o velmi zdařilé reporty, v některých částech game designu zbytečně rozsáhlé, ale autorům to pomohlo v uspořádání si myšlenek. Důležitými prvky v reportu jsou:

- Struktura
  - co bylo uděláno
  - co budeme dělat v následující době
  - kolik nám to prozatím zabralo
- Rozdělení na postup v rozdílných částech práce -- viz následující seznam
  - Odhad procentuálního postupu v jednotlivých kategoriích je nutnou součástí reportu
- SCREENSHOTY !!!
- V práci do budoucna je přehledně vidět, co bylo již splněno a co zůstává dál

## Každý report popíše a vyhodnotí postup prací v následujících kategoriích

### Game design

Jde o položky mající vztah k samotnému konceptu hry, tj. typy postav a NPC, lokace v mapě, druhy zbraní, úkoly, scénář levelu - nejde o implementaci, ale o rozmyšlení a návrh toho, jak to bude. V podstatě by tato položka měla být splněná minimálně na 75% po odevzdání Milestone 2 - Design Dokument.

### Engine

Obslužné skripty důležité pro samotný běh hry. Načítání modelů a textur, tvorba stromu scény, pohyb kamery, ovládání, UI, menu, shadery...

### Gameplay

Skripty mající přímý vliv na hru a její logiku. Tj. AI a chování NPC, bojový systém a systém odměn, postup hráče, inventář, a další chování hry.

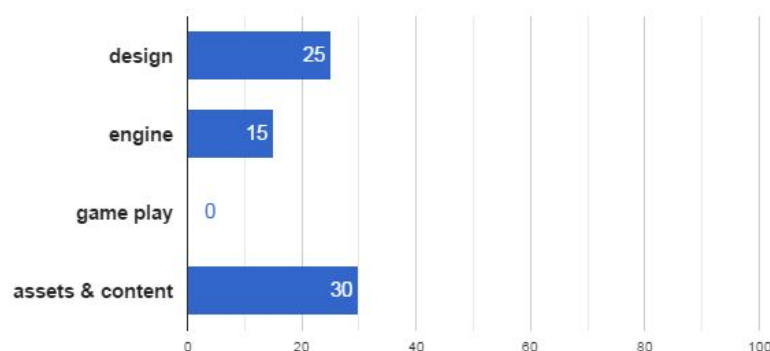
### Assets

Jde o tvorbu veškerého multimediálního obsahu jako jsou modely pro obecné objekty, postavy a NPC, modely map a lokací, textury, zvuky, animace.

# Report 1

Stav: 17.5%

## Project Gliese 832c



### Akutální stav hry

#### Game design

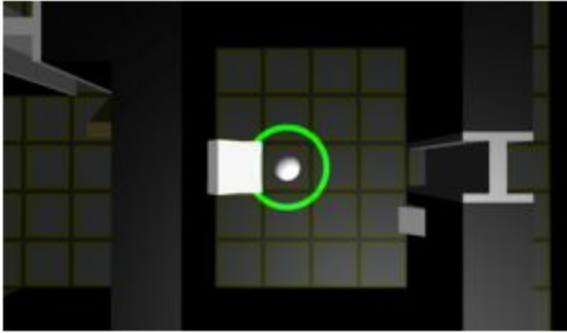
Základní vize příběhu, tak jak je popsána v konceptu, se prozatím nezměnila. Je vymyšlený úvod hry a zasazení hráče do děje. Úvodní mise by mohla být tutoriálem, kterýby měl za úkol hráče seznámit s ovládáním a mechanismy hry.

Zatím je navržena testovací mapa (viz Obr. 3), určena na testování enginu a jednoduchých mechanismů hry.

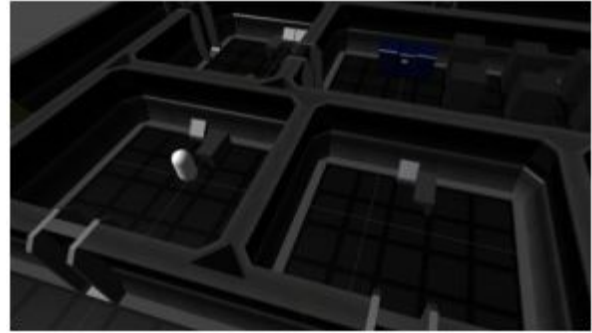
#### Engine

Engine zatím obsahuje základní prvky ovládání. Již je možné vybrat postavu a je možné hýbat kamerou. Volba vhodného ovládání je velmi důležitá, proto by bylo rozumné to trochu více rozepsat.

Způsobovládání kamery a selekce postav je mnoho, proto jsme se snažili být při výběru velmi pečliví. Nejprve je podstatné si uvědomit, co se na scéně bude dít a jak rychle je nezbytné aby hráč reagoval. V našem případě se na scéně objeví pár přepínatelných postav, z nichž každá může plnit určitý úkol. To znamená, že naše postavy jsou mnohem méně závislé než např. vojáci ve hře Starcraft. Proto jsme se rozhodli výběr zjednodušit a omezit ho pouze na jednu aktivní postavu (výběr postavy je ukázán na Obr. 1). Díky tomu, že máme v danou chvíli pouze jednu aktivní postavu, můžeme jí přizpůsobit i kameru, která v případě, že je libovolná postavu aktivní, tuto postavu sleduje. Zároveň absence vícenásobného výběru nám uvolnila levé tlačítko myši a tažení (běžně obeléníkový výběr), což jsme využili pro pohyb kamery. Tento pohyb je velmi dobře znám např. z nejrůznějších elektronických map, cožby mohlo do ovládání přinést větší intuitivnost.



Obr. 1: Snímek ze hry s aktivní postavou



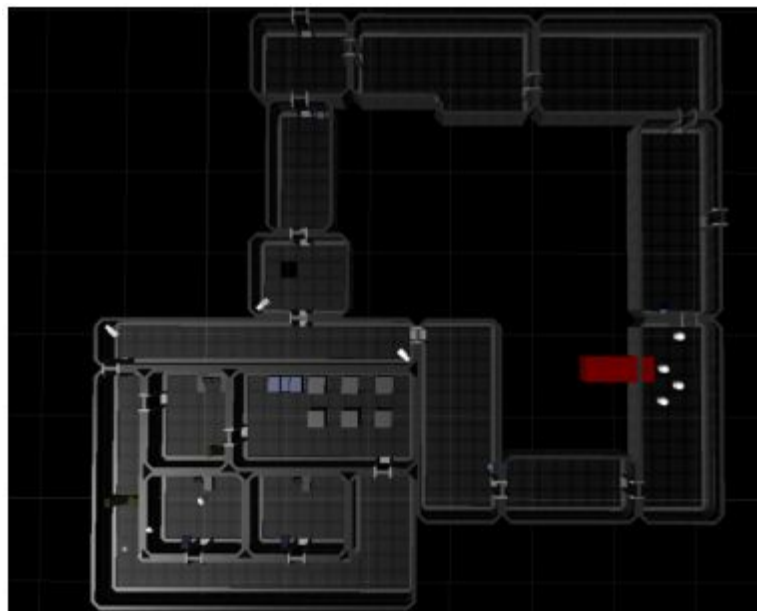
Obr. 2: Ukázka designu

## Gameplay

Prozatím nebyla implementována žádná funkcionalita.

## Assets

Vymyslet vhodný grafický styl je základem celé hry, od které se odvíjí způsob podání příběhu i hratelnost. Zatím máme rozvržené základní stavební prvky mapy, jako jsou např. zdi, dveře a podobně. Nic ovšem není finální. Ukázka designu je zobrazena na Obr. 2.



Obr. 3: Testovací mapa

## Práce do budoucna

Do budoucna je ještě mnoho práce. Níže jsou vyjmenovány úkoly s nejvyšší prioritou.

### Game design

- navrhnout postavy a jejich příběhy
- navrhnout stručný scénář obsahující jednotlivé herní kapitoly
- navrhnout základní typy místností

- navrhnout základní rozvžení oblastí na mapě(můstek, ubytovací části, strojovna, ...)

## Engine & Gameplay

- implementovat pohyb postav - navigace na mapě
- implementovat možnost interakce s okolím
- implementovat základní gui

## Assets

- vytvořit prototypy místností
- vytvořit postavy (přátelské i nepřátelské)
- vytvořit interaktivní prvky mapy (ovládání dveří, počítače, ...)
- navrhnout základní gui

## Hodinová dotace

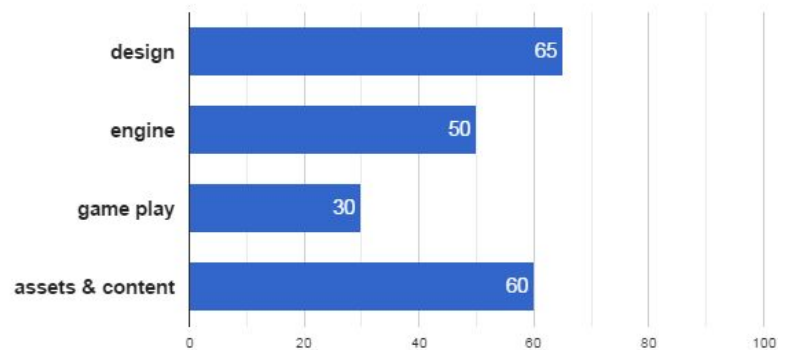
Vojtěch Micka: 16h 12min

Martin Kopřiva: 12h

## Report 2

Stav: 51.25%

# Project Gliese 832c



## Akutální stav hry

### Game design

Již je navržena základní verze příběhové linie. Jelikož celá hra probíhá na jedné mapě, jedná se prakticky o drobné úkoly, které hráč postupně plní. Ze začátku je spíše veden, ale postupně se mu dává větší volnost.

První část začátku: (Level design v podstatě kompletní)

*Hráč ovládá na začátku jednu postavu, je mu napovídáno myšlenkami dané postavy. Ta nemá tušení co se stalo.*

Prvním úkolem je zapnout backup generátor.

*Otevře si dveře, může si všimnout suti v chodbě a zapne backup generátor v servrovně.*

Druhým úkolem je zjistit co se děje.

*Vstoupí do security roomu, kde objeví mrtvého strážného pod hromadou suti, chce danou událost ohlásit na můstek a zjistit co se děje. Vezme si jeho key card. Zjistí, že nefunguje propojení s můstkem.*

Třetím úkolem je dostat se k zařízení přes které je daná část propojená s můstkem.

*Hráč zjistí pomocí kamerového systému, že přímá cesta je zablokováná dírou do vesmíru v chodbě. Musí proto jít oklikou. Dojde k dalšímu security roomu, přes kamerový systém zjistí, že v další místnosti jsou tupí nepřátelé. Leknutím omylem spustí alarm. Tím přiláká nepřátele k sobě, ti se prokoušou dveřmi a začnou ničit dveře do security roomu, otevře dveře do místnosti ve které je díra*

*do vesmíru a zabije všechny nepřátele. Zavře dveře a dojde k zařízení propojující s můstkem, začne opravovat.*

Druhá část začátku:

Hráč ovládá skupinu dvou vojáků v sousední sekci lodě. Tito vojáci jsou zavření v místnosti a mají pouze pistole. Vědí částečněco se děje/stalo.

Hlavní úkol(a úkol až do konce hry): zachránit přeživší a utéct z lodi pomocí střední průzkumné lodě(letadlo může mít mini warp pohon či co) a varovat další lodě ze Země mířící sem či varovat zem o flotile mířící na Zem(obyvatelé planety Gliese mohli přilet lodí vyhodnotit jako útok) pomocí vyslání signálu nebo vystřelení speciálního vysílacího satelitu.

*Za dveřmi v přilehlé chodbě slyší zvuky. Hráč použije schopnost taunt, nepřátelé se prokoušou dveřmi které nešlo otevřít. Jde o slabou variantu tupých nepřátel. Vojáci je zlikvidují a postupují dále, v přilehlé místnosti zabijí nepřátele a osvobodí skrývajícího se technika(ten může být napaden a hráč tak může otestovat druhou vlastnost taunt - naggrování nepřátel). Díky němu se dostanou do místnosti s drouny a zmocní se lehce poškozeného posledního funkčního scout/transport drona. Vědí, že přilehlá zbrojnice byla zasažena a obsahuje díru do vesmíru. Pošlou tedy drona do této zbrojnice, zjistí že poslední co zbylo je bedna s SMG/lepšími zbraněmi nežpistole, pomocí dronu tuto bednu dotáhnou ven a uzavřou dveře do zbrojnice. Ozbrojí se lepšími zbraněmi. Dojde k hangáru a zjistí(napřpomocí drona), že bylo vše zničeno a velká část hangáru chybí. Rozhodne se tedy, že se pokusí dál hledat přeživší a dostane se do místnosti kde byli nepřátelé, kterých se zbavil v první části. Následněse dostane k postavěs kterou začal. Konec tutoriálu?*

Zároveň jsou již navrženy třídy postav:

#### **Technik**

Dokáže opravovat vybavení. Dokáže otevřít některé dveře zásahem do systému dveří.

#### **Programátor**

Dokáže se hacknout do bezpečnostních systémů- kamerový systém a door control a rozjet napřpropojení s můstkem lodi.

#### **Voják**

Dobryv používání zbraní, může používat turrety a těžké zbraně. Má schopnost taunt(vytvoří hluk a možná i donutí nepřátele aby útočili na něj místo ostatních), kterou může nalákat nejbližší nepřátele. Umí řídit drony??

#### **Civilista**

Umí chodit.

Mapa

Mapa již dostává tvar, základní úkoly jsou narženy a mapa je podle nich přizpůsobena. Aktuální mapa je zobrazena na obrázku 1.

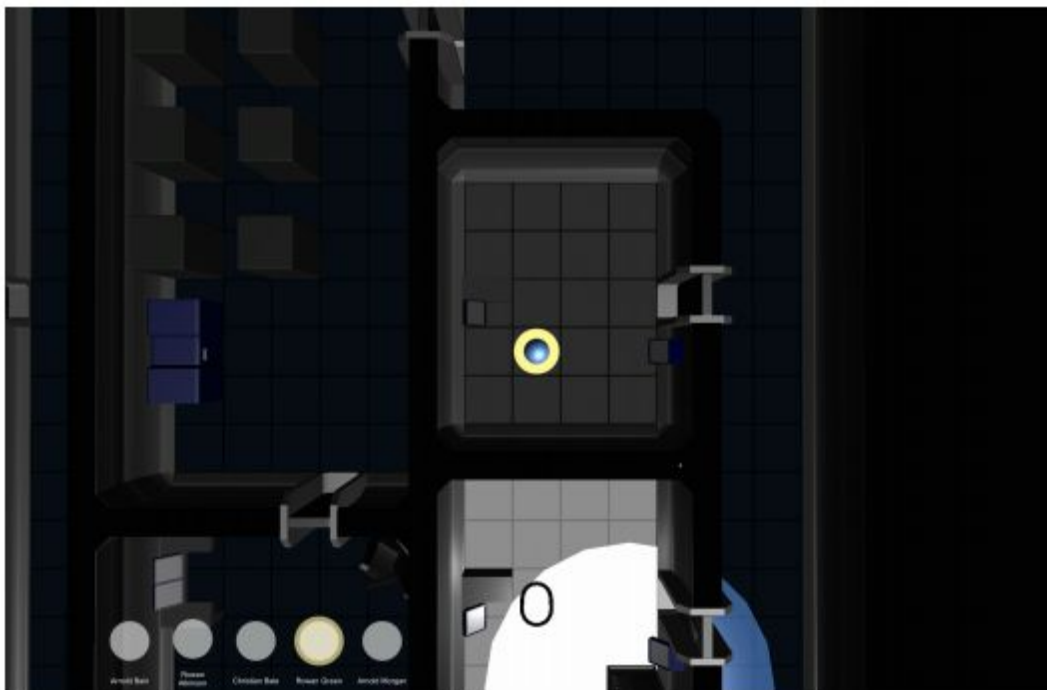


Obr. 1: Aktuální mapa

## Engine

Engine z velké části zůstal stejný, pouze se upravovalo několik základních funkcionalit, které se hráči přímo neprojeví. Měnil se např. systém označování postav, nyní je jednoduše možné přepínat postavy máme-li referenci na její GameObject.

Nejvíce času zabralo UI - nyní je funkční HUD, kde si hráč může přepínat postavy, které jsou na mapě dostupné. Aktuální herní obrazovka s novým HUDem je vidět na obrázku 2.

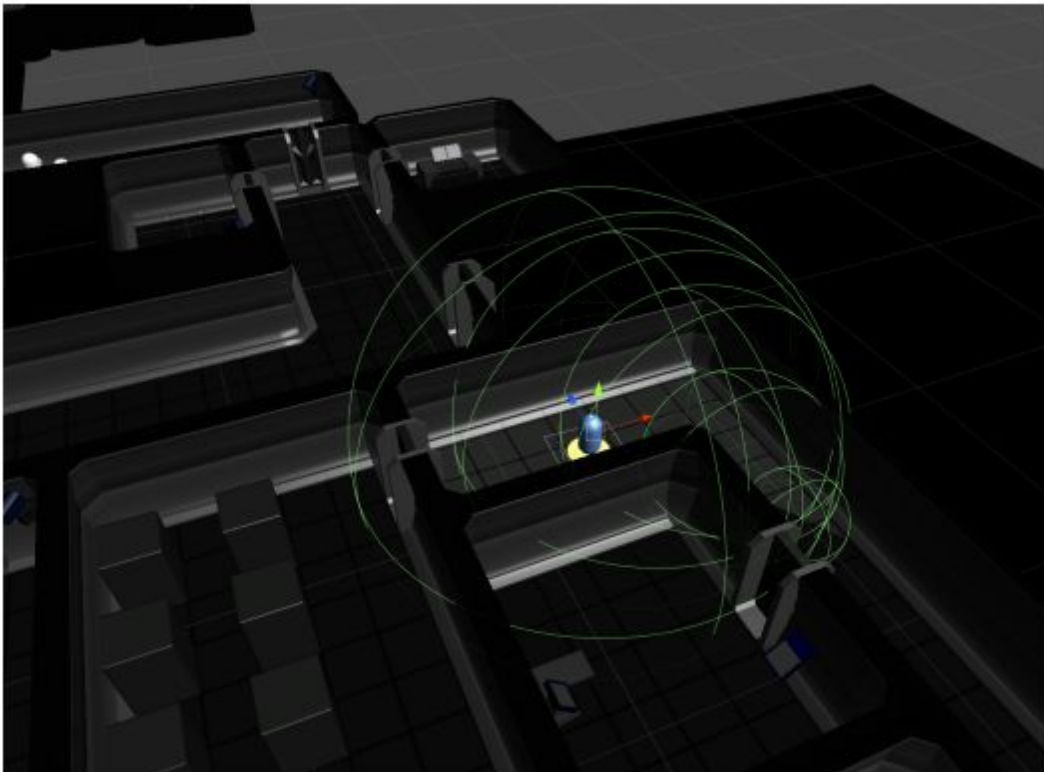


Obr. 2: Ukázka ze hry

## Gameplay

Vytvořili jsme základ nepřátelského AI.

Asi největší novinkou je systém šíření zvuku (viz obárzek 3). Každá postava vydává při pohybu zvuk, který může nepřítel zachytit. Jakmile nepřítel zachytí zvuk, může odhadnout z jakého zdroje a z jaké pozice tento zvuk přišel a prozkoumat pozici zdroje zvuku. Každý zdroj zvuku má různé parametry jako rychlost šíření, počáteční intenzita a podobně, díky tomu jsme schopni snadno simulovat různá prostředí a různé typy zvuků. Prozatím se zvuk vytváří pouze při pohybu postav a simuluje tím zvuk kroků, intenzita je závislá na povrchu (který je zatím všude stejný) a na rychlosti pohybu.



Obr. 3: Šíření zvuku

## Assets

Přidalo se pár nových modelů (server, záložní generátor, ...). Zároveň se navrhl základní HUD. HUD bude velice jednoduchý, bude obsahovat prakticky pouze zkratky k dostupným postavám a nějaké základní informace o nich, jako např. životy, ty budou znázorněny pravděpodobně změnou barvy, případně nějakým jednoduchým health barem. Aktuální HUD je vidět na obrázku 2 dole.

## Práce do budoucna

Do budoucna je ještě mnoho práce. Níže jsou vyjmenovány úkoly s nejvyšší prioritou.

## Game design

- navrhnout postavy a jejich příběhy



- navrhnout stručný scénář obsahující jednotlivé herní kapitoly
- navrhnout základní typy místností
- navrhnout základní rozvření oblastí na mapě (můstek, ubytovací části, strojovna, ...)

## Engine

- implementovat základní gui
- implementovat menu

## Gameplay

- implementovat pohyb postav – navigace na mapě
- implementovat možnost interakce s okolím
- implementovat nepřátelské AI

## Assets

- vytvořit prototypy místností
- vytvořit postavy (přátelské i nepřátelské)
- vytvořit interaktivní prvky mapy (ovládání dveří, počítače, ...)
- navrhnout základní gui
- navrhnout menu

## Hodinová dotace

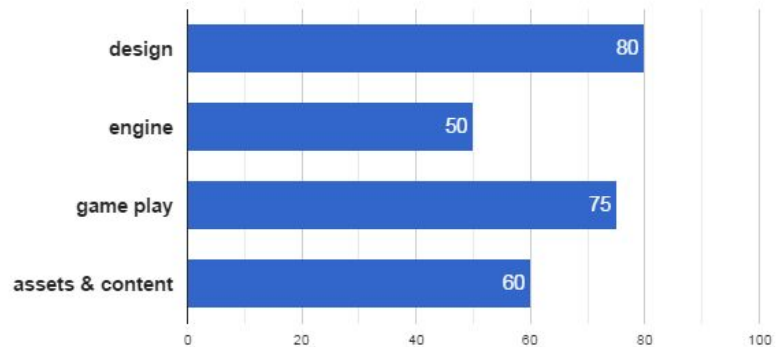
Vojtěch Micka: 42h 12min (+26h)

Martin Kopřiva: 34h (+22h)

## Report 3

Stav: 66.25%

# Project Gliese 832c



## Akutální stav hry

### Game design

Již jsme konečně dali dohromady jednotlivé parametry objektů a postav.

#### Třídy postav

**Mechanik** - dokáže opravit rozbité systémy, je schopen otevřít některé druhy dveří, umí řídit drony, dokáže se bránit

**Programátor** - dokáže se nabourat do systémů až střední úrovně zabezpečení

**Voják** - umí používat zbraně a turety, má schopnost taunt(vytvoří hluk a donutí nepřátele aby útočili na něj místo ostatních), kterou může nalákat nejbližší nepřátele

**Kolonista** - je schopen chodit

**Droni** - dálkově řízení roboti, umožňující bezpečněprozkoumávat riskantní části lodi a lze pomoci nich transportovat těžké náklady z míst bez podpory života

Třída postav	Počet životů	Typ útoku	Síla útoku
Mechanik	100	Meele	15
Programátor	100	—	—
Voják	200	Meele / Ranged	30 / Podle zbraně
Kolonista	80	—	—
Droni	800	—	—

## Nepřátelé

Nepřátelé jsou schopni rozbít dveře. Nepřátelé jsou nedílnou součástí hry a hráč je tedy bude potkávat často. Ve hře lze potkat dva druhy nepřátel:

**Rippers** - agresivní nepřítel schopný boje nablízko, bez rozmyšlení se vydá za jakýmkoliv zvukovým podnětem

**Gises** - inteligentní opatrný nepřítel vybavený střelnou zbraní, schopný velet Ripperům ve své blízkosti

Druh nepřitele	Počet životů	Typ útoku	Síla útoku
Ripper	40 / 80 / 120	Meele	20 / 40 / 60
Gise	80	Ranged	30

## Zbraně

Vojáci se skrze level mohou vybavit různými zbraněmi. Zbraně mají stanovený počet nábojů v zásobníku a omezený počet zásobníků, které voják dokáže unést

Typ zbraně	Zásobník	Síla útoku na střelu
Pistole	12	10
SMG	46	5
Útočná puška	32	10
Mínigun (připevněný)	500	20

## Interaktivní objekty

Hra obsahuje množství interaktivních objektů. S těmi může hráč interagovat skrze jednotlivé postavy, pokud tyto postavy splňují požadavky pro interakci.

Druh objektu	Způsob interakce	Vyžadovaná třída postavy
Dveře	Přímá / Pomocí dveřního terminálu	Mechanik / —
Dveřní terminál	Přímá	Všechny mimo drone
Dveřní terminál zamčený	Přímá	Programátor
Kamera	Pomocí kamerového terminálu	—
Kamerový terminál	Přímá	Všechny mimo drone
Kamerový terminál zamčený	Přímá	Programátor
Alarm	Pomocí bezpečnostního terminálu	—
Hlukotvorné zařízení	Přímá	Mechanik
Záložní generátor	Přímá	Všechny mimo drone
Záložní generátor poškozený	Přímá	Mechanik
Přípevněná zbraň	Přímá	Voják
Bedna	Přímá	Voják / Drone
Zbraň	Přímá	Voják
Bezpečnostní karta	Přímá	Všechny mimo drone

## Mapa

Mapa se bude zřejmě předělávat na základě konkrétního layoutu lodi, který vymýšlíme, zatím se ale příliš nezměnilo.

## Engine

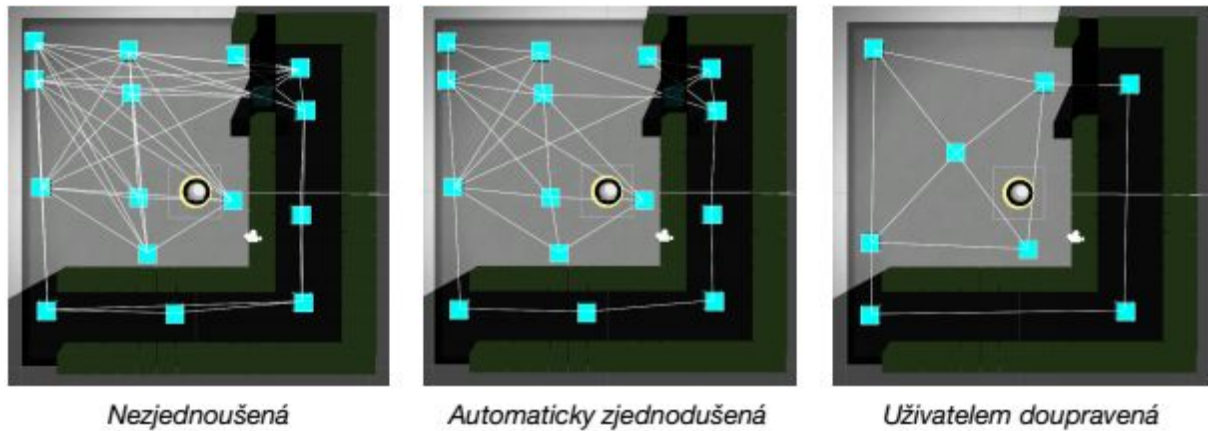
Neproběhla žádná aktualizace.

## Gameplay

Aktuálně jsme nejvíce pracovali na **Enemy AI**, k tomu bylo potřeba vytvořit síť waypointů, podle které by se nepřátelé mohli pohybovat. Základ je v tom, že nechceme aby nepřátelé hloupě sledovali jednu či více tras, ale spíše chceme aby prozkoumávali loď. To znamená, že bychom museli postavit síť waypointů pro celou mapu, proto jsem vytvořil offline generátor waypointové sítě, která automaticky z NavMeshe vytvoří celou síť, která se poté snadno upraví podle našich představ.

### Síť waypointů

Generátor funguje velmi jednoduše, nejprve vytvoří waypointy ve všech vertexech NavMeshe a poté se je snaží rozdělit do clusterů pomocí metody KNN, výsledné waypointy jsou pak středy těchto clusterů, díky tomu se síť znatelně zjednoduší a např. v chodbách dostaneme waypointy krásně uprostřed. Další částí je nalezení sousedů, k tomu používáme rychlý NavMesh.Reycast. Nakonec je potřeba sousedy prořezat, jinak by byla síť zbytečně komplikovaná. Na obrázku 1 je vidět různě vygenerovaná síť v jednoduché mapě. Takto vytvořená síť se dá použít k mnoha věcem, celkem zatím uvažujeme o třech: **pravděpodobnost výskytu hráče, šíření kyslíku a šíření pachu hráče.**



Obr. 1: Ukázka generování sítě waypointů

#### Pravděpodobnost výskytu hráče

Ve chvíli kdy je hráč spatřen nepřítelem, nepřítel se snaží sledovat hráče. Nepřítel v průběhu sledování ukládá do hráči nejbližšímu waypointu pravděpodobnost výskytu = 1, ve chvíli co se hráč ztratí, pravděpodobnost výskytu se propaguje s daným poklesem do sousedních waypointů, díky tomu má nepřítel přehled o tom, kam by hráč mohl uniknout a může prozkoumávat místa s pravděpodobností  $> 0$ . Zatím se jedná pouze o nápad, jehož implementace není zcela prioritní, ale v každém případě přispějí zajímavější hratelnosti.

#### Šíření kyslíku

Každá místnost v lodi musí být naplněna okysličeným vzduchem, tak aby hráč mohl přežít. V případě, že by se např. vytvořila v trupu lodi díra, začal by vzduch z lodi unikat ven, což by se pomalu začalo šířit do všech ostatních propojených míst lodi. Toto šíření se dá opět simulovat na síti waypointů, stejně jako následné uzavření díry a opětovné šíření vzduchu.

V aktuální verzi je šíření vzduchu udělané přímo v místnostech. Každá místnost má maximální kapacitu vzduchu a aktuální kapacitu vzduchu, která se mění v závislosti na vzduchu v okolních místnostech.

#### Šíření pachu hráče

Jeden typ nepřátel by měl být schopen zachytit pach hráče a následně sledovat gradient pachu až by došel ke zdroji. Tento fenomén se dá opět simulovat na síti waypointů podobně jako šíření vzduchu.

#### Zvuk

Již dříve jsme experimentovali se šířením zvuku, v minulém reportu jsme popsali verzi ve které je zvuk šířen pomocí SphereColliderů, které jsou generovány přímo ze zdroje zvuku. Po dlouhém uvažování jsme se rozhodli tento přístup změnit na klasičtější, ale především mnohem efektivnější. Každá postava, která je schopna zachytit zvuk, má kolem sebe SphereCollider, který detekuje objekty, které jsou slyšitelné a sama si dopočítá jestli je zvuk schopna slyšet v daném prostředí či nikoliv. Tato metoda není tak elegantní, ale funguje mnohem rychleji, protože se nemusí neustále vytvářet nové objekty.

## Assets

Neproběhla žádná aktualizace.

## Práce do budoucna

Do budoucna je ještě mnoho práce. Níže jsou vyjmenovány úkoly s nejvyšší prioritou.

### Game design

- navrhnout postavy a jejich příběhy
- navrhnout stručný scénář obsahující jednotlivé herní kapitoly
- vymyslet způsoby zakončení (podmínky světa)
- navrhnout základní typy místností
- navrhnout základní rozvžení oblastí na mapě (můstek, ubytovací části, strojovna, ...)

### Engine

- implementovat základní gui
- implementovat hlavní menu

### Gameplay

- implementovat pohyb postav – navigace na mapě
- implementovat možnost interakce s okolím
- implementovat nepřátelské AI
  - volný pohyb nepřátel
  - vjemy (sluch a vidění)
  - útok
- implementovat jednotlivé postavy
- implementovat kamery a drony
- implementovat interakce se všemi interaktivními objekty

### Assets

- vytvořit prototypy místností
- vytvořit postavy (přátelské i nepřátelské)
- vytvořit interaktivní prvky mapy (ovládání dveří, počítače, ...)
- navrhnout základní gui
- navrhnout menu

## Hodinová dotace

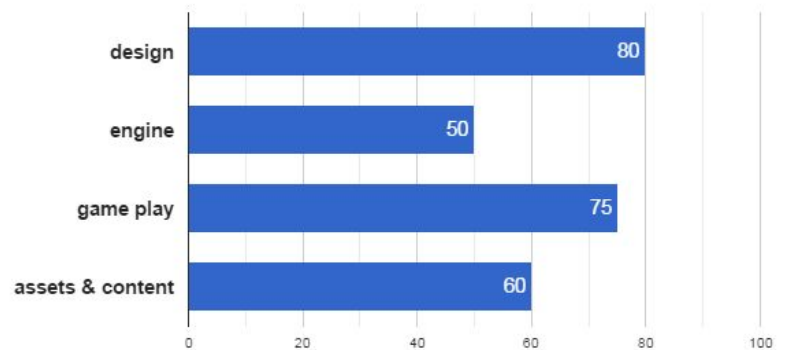
Vojtěch Micka: 74h 12min (+32h)

Martin Kopřiva: 58h (+24h)

## After action report

Stav: 87.8%

## Project Gliese 832c



### Akutální stav hry

V každé kategorii sepsat co je vskutku implementováno a co jste vynechali a proč, co jste podle vás udělali dobře a co špatně. Není nutností aby procentuální plnění bylo 100% - jde nám o sebekritické zhodnocení stavu hry. Na bodové ohodnocení nemá vliv "splněnost" pokud došlo k posunu od posledního reportu, ale kvalita tohoto reportu.

#### Game design

#### Engine

#### Gameplay

#### Assets

### Závěr

Kompletní shrnutí implementace hry - celkový pohled na práci, zhodnocení spolupráce, objektivní zhodnocení celé hry vzhledem ke konceptu, atd.

### Hodinová dotace

Vojtěch Micka: 110h (+36h)

Martin Kopřiva: 70h (+12h)