Alterborne

Av Malin Sommar, Simon Bengtsson, Lydia Marcondes, David Furby

Om spelet | Sida 1

Veckoplanering och ansvarsområden | Sida 2

Vad hade vi utvecklat om vi haft mer tid | Sida 8

Struktur på spel och hemsida | 8

Backlog | 9

Våra tankar om arbetet | 10

CMS | 11

Säkerhet | 11

**Om spelet**

Alterborne är ett gammaldags turn-based rpg spel (role-player game). Du har ett party på 4 pers.

Warrior, som är tankig och kan buffa lagkamrater eller debuffa fiender. En ranger, som kan gömma sig i skuggorna för att göra extra skada. En mage som har hög skada men väldigt lågt Hp. Healer som har låg skada men kan hela lagkamrater.

Alla klasser har olika skills men alla klasser har en vanlig attack och block, dessa kostar 2 energi. Skills kostar oftast mer men är användbara.

Alla klasser har olika mycket Hp, damage och block. Dessa höjs ifall du går upp i level eller hittar bättre gear. Vapen och armor kan du få efter fights eller i köpa i shopen, där du också kan få eller köpa potions.

Det finns 4 olika sorters potions, healing potion, block potion, energy potion och strength potion. Dessa finns i minor, lesser och major.

När du kommer in i världen kan du välja vart du vill gå, alla fight ingångar har en rekomenderad level bredvid sig men du kan gå in i dem när du vill. Alla fights har en vanlig fiende man brukar möta, men det finns också en chans att möta banans boss. Dessa är mycket starka men ger bra loot.

I världen hittar du också shopen där du alltid kan fylla på med potions eller gear.

Spela och se ifall du kan rädda världen!

**Veckoplanering och ansvarsområden**

Lydia:

Hemsida, databas.

* Vecka 1: Planering av hemsidans innehåll och design, undersökning av JFrame
* Vecka 2: Undersökning av Swing/JFrame, började jobba på hemsidan
* Vecka 3: La till tabeller i hemsidan, styling
* Vecka 4: Styling och UX-design
* Vecka 5: CSS/webbdesign
* Vecka 6: Styling, undersökning av databasrelaterade saker
* Vecka 7: Försöka få en nested dropdown i navbaren jag hade gjort
* Vecka 8: Ny navbar med vanlig dropdown, databas
* Vecka 9: Få databasen att funka, sista ändringar på hemsidan

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

David:

Overworld/worldmap

V1. Börjar arbetet med att skapa en healthbar till stridssystemet, men väljer snabbt bort den mot att jobba på en egen del av spelet i formen av en OpenWorld-karta. Hittade en till synes väldigt bra youtube-guide för hur man bygger upp en sådan, och ska spendera de första veckorna med att studera den. De jag hittills lyckats göra med det är att manipulera rutan att visa vilda pixlar genom en kombination av Threads, grafik-object, tick och render-metoder. Har också pluggat en hel del Swing

V2. Jobbar nu med att skapa en SpriteSheet-bild. varje 8x8 ruta kan representera ett mönster för hur hur en av OverWorld:ens block kan se ut. Genom att använda ett imageIO-object läste java av bilden och började räkna ut rutans position inom ramen. Vilket atm resultera i att mönstret som formats upprepas över hela ramen. Fick hjälp från Ulf när bilden inte såg ut som den skulle

V3.Strul med Git hela veckan.

Kameran går nu att flytta genom InputHandler-klassen. Men när man gör det ritas inga nya bilder.

Youtube-serien är inte jättebra på att beskriva va det gör, vilket resulterar i att man lätt missar saker eller att man inte förstår. Så viktigt att skriva mycket kommentarer

V4. Fortfarande strul med Git.

Finns en gubbe man kan flytta runt på skärmen. Ritade en enkel figur över fyra 8x8 SpriteSheet rutor och satte ihop dem för att skapa den. Är satt rakt i mitten så att de ser ut som det är den som styr när kameran flyttar sig. Inga animationer atm.

V5. Mer strul med Git.

Animationer finns nu och en bana för gubben att gå runt på. En liten grön bana, med en blå yta vid sidan för att likna en sjö. Gubben Går på vattnet atm.

V6. Finns nu block med olika funktioner. Stenar går inte att gå igenom och gubben simmar genom vattnet. Ska börja göra min egna karta och bygga ett sätt att gå mellan olika min OverWorld och de olika striderna. Började skriva om hela projektet för att matcha MVC

V7. Lyckas inte koppla ihop de två delarna av spelet. Tanken va att de skulle finnas en slumpmässig chans att om man går på vissa ytor av banan så hamnar man i en strid. Min värld är under tiden pausad i bakgrunden, och när striden är klar så avslutas paus:en och man kan fortsätta gå runt igen. Men någonting hur threads funkar i kombination med resten av spelet får det inte att funka.

V8. Min grundläggande tanke om hur man ska hamna i strid skrotas. Nu istället går man in i olika portar som startar upp en strid. Min värld blir disposed() och startas istället som ny när striden är klar. Jag fortsätter också att utveckla kartan med nya brickor etc.

V9. Spelet är nu i princip klart. Har jag jobbat här och där med att förbättra kartan, men har också lagt till funktionaliteten att username som ska visas upp vid sin highscore också syns över min gubbe i världen. Förutom de är nu också alla strids-vägar namngivna och har ett en rekommenderad level för att möta

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Malin:**

Klasser som används:

MasterModel, LootController, LootView, UserNameController, UsernameView, ShopController, ShopView, LevelUpController, LevelUpView, LoseController, ForestFightController, ForestFightView, ForestBossFightController, ForestBossFightView, TutorialController, TutorialView, HubController, HubView

Klasser som ej används:

Healer, mage, warrior, ranger, FightModel, Loot/Shop/XPModel, inventory, Shop, Hub,

**V1:**

Läsa på om JFrame

Grund kod för Hub & new game (i nuläget: HubController, hubView).

Börjar med grundkod för alla fights.

Börjar på party-members klasser. (i nuläget warrior, ranger, mage, healer)

**V2:**

Fortsätter mer fight klass.

Fortsätter med party-member klasser.

Skapar vapen/armor kod för party-members.

Börjar på shop klass (I nuläget shopController, shopView).

**V3:**

Fortsätter med fight klass.

Shop klass klar.

Lägger till funktionalitet för potions.

Fortsätter på losescreen som simon påbörjat (i nuläget: LoseView)

**V4:**

Klar med fight grund, ger över till simon.

Påbörjar Xp system (i nuläget: LevelupController, levelUpView)

Importera egen font.

Route choice klass (ifall davids overworld inte skulle fungera, denna används ej).

Påbörjar lootScreen klass (i nuläget LootController, LootView)

**V5:**

Fortsätter med LootScreen

Påbörjar Xp system (i nuläget: LevelupController, levelUpView)

**V6:**

Bygger om all kod till MVC

**V7:**

Bygger om all kod till MVC

**Lov:**

Bygger om all kod till MVC

MasterModel klass

Rensar program på onödiga klasser.

Börjar fixa kopplingen mellan alla klasser för att på ett fungerande spel.

**V8:**

Fixar kopplingen mellan alla klasser för att på ett fungerande spel.

MasterModel metoder/uppbyggnad

Kodar i xp, loot, shop klasser etc.

**V9:**

MasterModel metoder/uppbyggnad

Fixar kopplingen mellan alla klasser för att på ett fungerande spel.

Loop i masterModel som håller igång spelet.

UserNameController, UserNameView

Fixar potions ordentligt

Log, JavaDoc, kommentarer, powerpoint.

BugFixes

**Simon:**

Klasser:

ForestFightController, ForestFightView, ForestBossFightController, ForestBossFightView, CaveFightController, CaveFightView, CaveBossFightController, CaveBossFightView, FieldFightController, FieldFightView, FieldBossFightController, CastleBossFightView, CastleFightController, CastleFightView, CastleBossFightController, CastleFightController, SwampFightController, SwampFightView, SwampBossFightController, SwampBossFightView,

MusicPick

V1:

Läs på om JFrame

Letar efter musik, bakgrunder, gifar, etc.

New Game frame (hubView)

V2:

Läs på om JFrame

Påbörjar MusicPick klass

Skriver namn till vapen och armor.

V3:

Börja utveckla animationer i testklass.

V4:

Fortsätter utveckla animationer

Tar över fight klass för att lägga in animationer och autobattle på fiender.

V5:

Fortsätter utveckla animationer

Fortsätter utveckla autobattle

V6:

Fortsätter utveckla animationer

Fortsätter utveckla fight klass, delar upp det i MVC.

Börjar utveckla spells.

V7:

Lägger in animationer

Arbetar i forestFight

Arbetar i caveFight

Fortsätter utveckla spells.

Lov:

Lägger in animationer

Fortsätter utveckla spells.

SpellDamage

Target system

V8:

Fortsätter utveckla spells.

V9:

Fortsätter utveckla spells.

Skapar FieldFight

Skapar SwampFight

Skapar CastleFight

**Vad hade vi utvecklat om vi haft mer tid.**

* Vi skulle vilja ha fler fights, tex mountain fight.
* Fler items som bomber och andra potions.
* Balansera spelet bättre.
* Meny/exit inne i spelet.
* Egna sprites, bakgrundsbilder och musik.
* I overworld, random encounter i skog, mountain etc. Som i pokemon.
* Nested dropdowns med hjälp av till exempel React på hemsida.
* Mer backstory för spelet och karaktärerna
* Hover box över Legendary/Epic/Rare/Common som förklarar vad de betyder på hemsida.
* Göra hela hemsidan responsiv

**Struktur av spel och hemsida.**

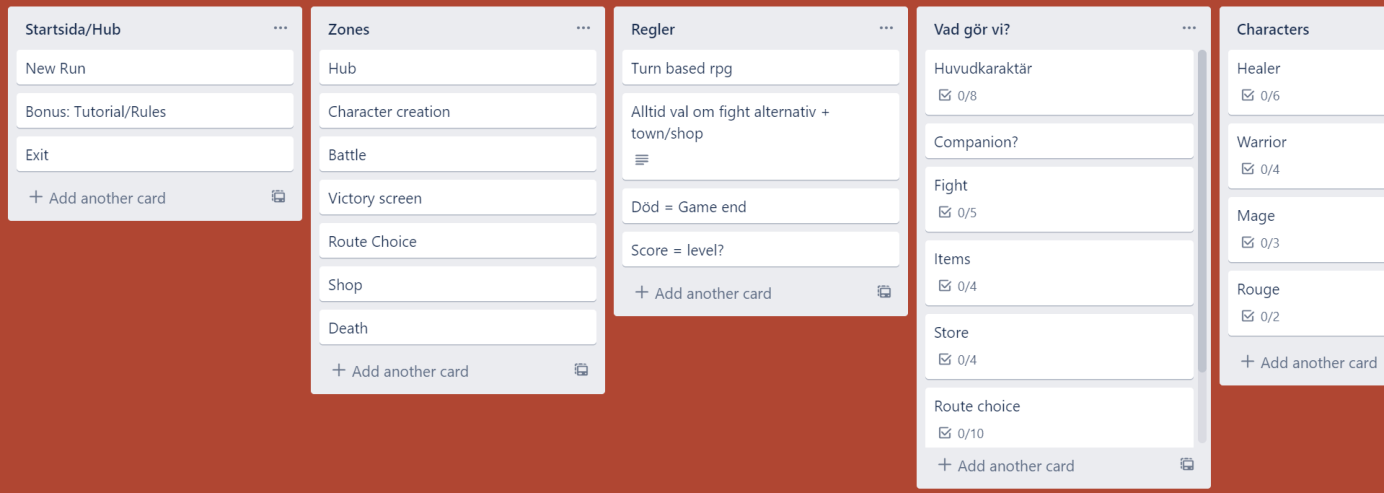
Vi använder oss av MVC i vårt spel. Vi har en MasterModel och nästan ingen annan model, bara controllers och views som mastermodel skickar data till.

Vi har sammanlagt 46 klasser så det skulle ta för lång tid att gå igenom alla, men nedan är en simpel flowchart på hur uppbygdnaden ser ut. Dem klasserna som inte är med är tex alla fightKlasser och flera klasser som tillhör WorldModel/WorldController/WorldView.

Se javadoc kommentarer för mer detaljerad information om varje klass.



**BackLog (Initiala vertionen)**



****

Vår första tanke är någorlunda lik hur spelet blev. Vi ville ha lite fler fights, items etc. Vi tänkte också ha fler stats som tex crit, evade och intelegence.

**Våra tankar om arbetet.**

**+**

Vi är alla nöjda med resultatet. Vi fick till det så som vi ville ha det trots tidspressen och brist på kompetens om swing och databaser.

Vi tyckte det var roligt, vi hade det trevligt i gruppen och därför blev det inte så jobbigt med motgångar.

Vi är nöjda med våran kommunikation, vi uppdaterade alltid varandra om vad vi gör, när vi uppdaterar git och när vi behöver hjälp med något.

Sist men inte minst så har vi utvecklats mycket under projektet, vi har lärt oss så mycket så snabbt att man oftast fått bygga om metoder för man kommer på nya sätt att bygga saker bättre.

**-**

Planeringen var inte på topp, vi satsade på extrasaker för tidigt och borde ha fokuserat på att få ihop ett spelbart spel tidigare. Vi borde även ha planerat med MVC från början.

På flera ställer har vi dublicerad kod, framförallt i striderna. Vi försökte fixa detta för ett tag sedan men kom inte på något bra sätt och nu har vi för ont om tid.

Vi var dåliga på att logga vad vi gjort i tid. Borde haft fler gruppmöten och loggat tillsammans.

?

Vi är fortfarande osäkra på hur databaser fungerar.

!

Vi har lärt oss swing väldigt bra under denna perioden.

Mer eller mindre förstått oss på hur MVC byggs upp.

Threading.

=

Vi är alla nöjda med resultatet och allas insatser.

**CMS**

**Fördelar med CMS:**

Man kan snabbt få ihop en fungerande hemsida. Bra om man har kort deadline.

Bra för privatpersoner eller småföretag som inte vill lägga mycket pengar på en programmerare för att göra en proffsig hemsida.

Kräver inga kod-färdigheter.

**Nackdelar med CMS:**

Man kan inte bygga lika avancerade program och kan lätt träffa en vägg där man inte längre kan utveckla hemsidan mer.

Kan vara svårare att sätta in säkerhet då du inte byggt koden själv och därför inte veta vart dem svaga delarna ligger.

**Säkerhet**

**Java**

* Eftersom vår databas bara är uppkopplad till localhost är vi väldigt säkra.
* Se till att använda private där det går så att det är svårare att komma åt klasser och metoder.
* Ha en login sida.

**Hemsida**

* Säkerhets checka cockies så att hackare inte lätt kan ta sig in på någon annans instans.
* Det finns massa plugins man kan använda och html sanitizers.