

SkilledTD

Multiplatform Development Pepijn Thijssens

Concept

SkilledTD is een tower defense, waarbij de mobs van punt A naar B gaan, niet in een vast pad maar ze zoeken de kortste route naar het einde. Je plaatst torens waarmee je een zo goed mogelijke route moet maken. De twist zit hem erin dat torens plaatsen per platform anders is. PC build kan je simpel weg de positie aanklikken met je muis. Op Android loopt de X en daarna de Y coördinaten waarbij je op het juiste moment op het 'one button scherm' drukt waarmee je de coördinaat vast zet. Zie figuur 1.



(Figuur 1. Gameplay)

Proces

De bedoeling was eerst om het te bouwen in Unity3d waarbij de desktop build 3d graphics utiliseert en de android variant 2d graphics. Na het toevoegen van pathfinding van de mobs, werd het spel al veels te traag voor m'n pc om te runnen in een groot grid (12x24). Nog niet eens getest in android omdat m'n build nog helemaal in 3d was. Toen ben ik hiermee gestopt.

Het spel is gebaseerd op een concept wat ik in het verleden al een keer half geïmplementeerd had in java om mezelf java/android development aan te leren. Dit project had ik weer opgepakt omdat deze code veel efficiënter was en al een werkende build op m'n oude samsung s4 had. Hierbij had ik nog wat bugs eruit gehaald en de code wat robuuster gemaakt.

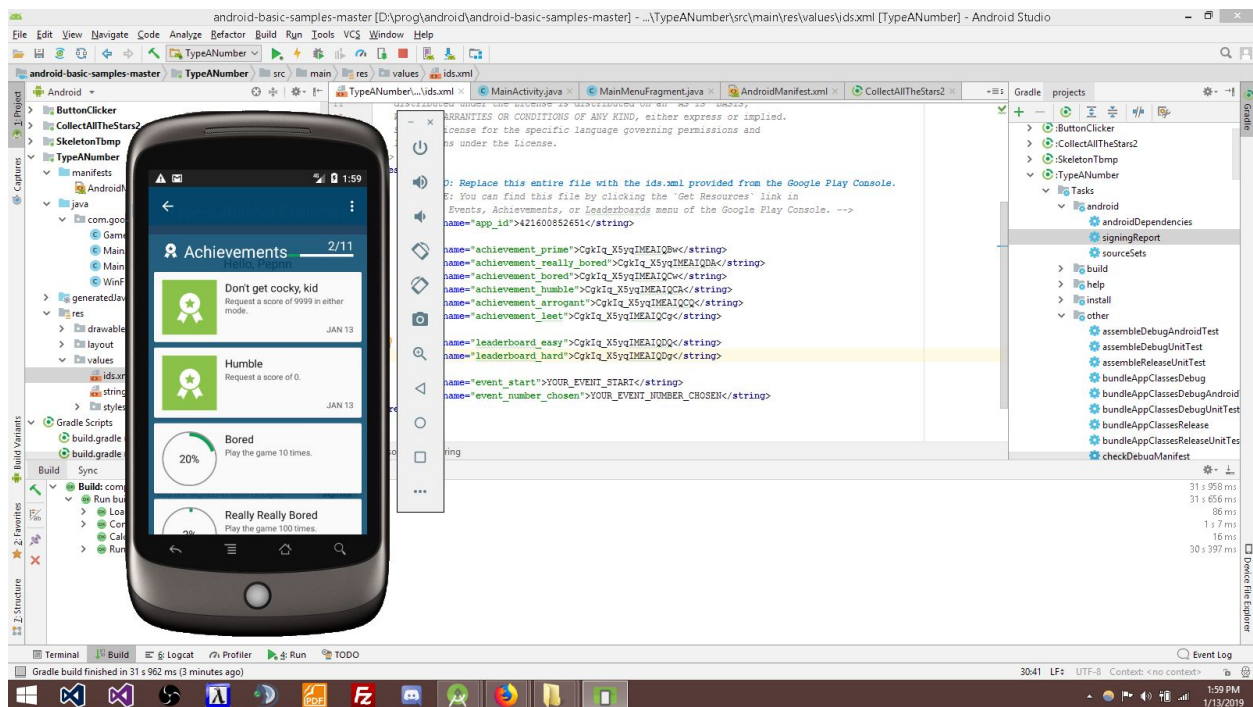
Toen was het idee om aan dit spel een achievement systeem toe te voegen. Gekeken naar Steamworks voor de desktop build, maar zit achter een 100 euro paywall. Google Play ben ik gaan gebruiken voor de android variant. Na dagen proberen de android SDK/gradle versie/Android Studio allemaal op de juiste versie te krijgen om Google Play werkend te krijgen.

Ik heb een van de Google Play achievements werkend gekregen na veel moeite. Dus mijn app kan connecten met m'n Google Developer Console. Alleen lukte mij het niet om in zo'n korte tijd nog alles uit te vogelen hoe het precies zat met het opvragen en pushen van alle data en dat te combineren met m'n libgdx library. Libgdx is een game oriented graphics library voor java.

<https://developers.google.com/games/services/android/achievements>

<https://developers.google.com/games/services/common/concepts/achievements>

Toen heb ik snel een AchievementSysteem uitgedacht maar niet uitgewerkt waarvan resten nog in de github te vinden zijn. Dit heb ik niet uitgewerkt omdat mn PlayStore achievements te complex waren. Toen maar voor gekozen om met een simpele interface InputHandler die iedere build variant implementeert en aangeeft aan het spel wat hij moet laten zien in de Input methods iedere loop. (PC variant niks, android build de momentele X en Y coördinaten die desbetreffende vakjes kleuren).



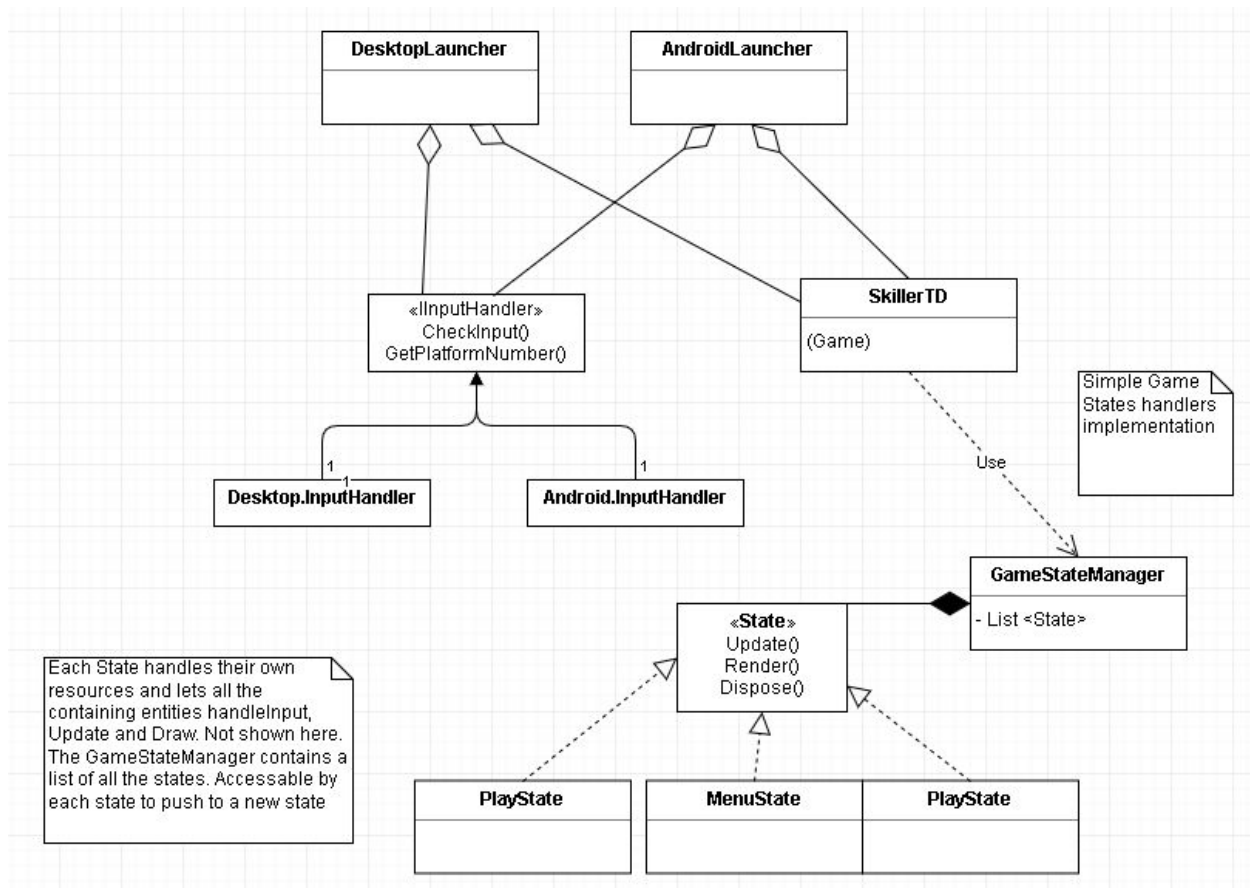
(Figuur 2 Android PlayStore Achievement werkend voorbeeld)

MPD & UML

Voor het ontwikkelen van apps, gebruikt android studio gradle. Gradle is een build tool waarbij voor ieder verschillend platform je verschillende build scripts schrijft die dan de juiste resources,

code files, dependencies en sdk gebruiken en alles goed in elkaar zet.

<https://developer.android.com/studio/build/>



Extra Bronmateriaal

https://www.researchgate.net/publication/273455882_Achievement_Systems_Explained

<http://swarmconnect.com/admin/docs/libgdx> (ander achievement systeem niet gebruikt)

<https://gamedevelopment.tutsplus.com/tutorials/how-to-code-unlockable-achievements-for-your-game-a-simple-approach--gamedev-6012> (AchievementSysteem niet geïmplementeerd)