

# DTABANK

*Manage your data and users*

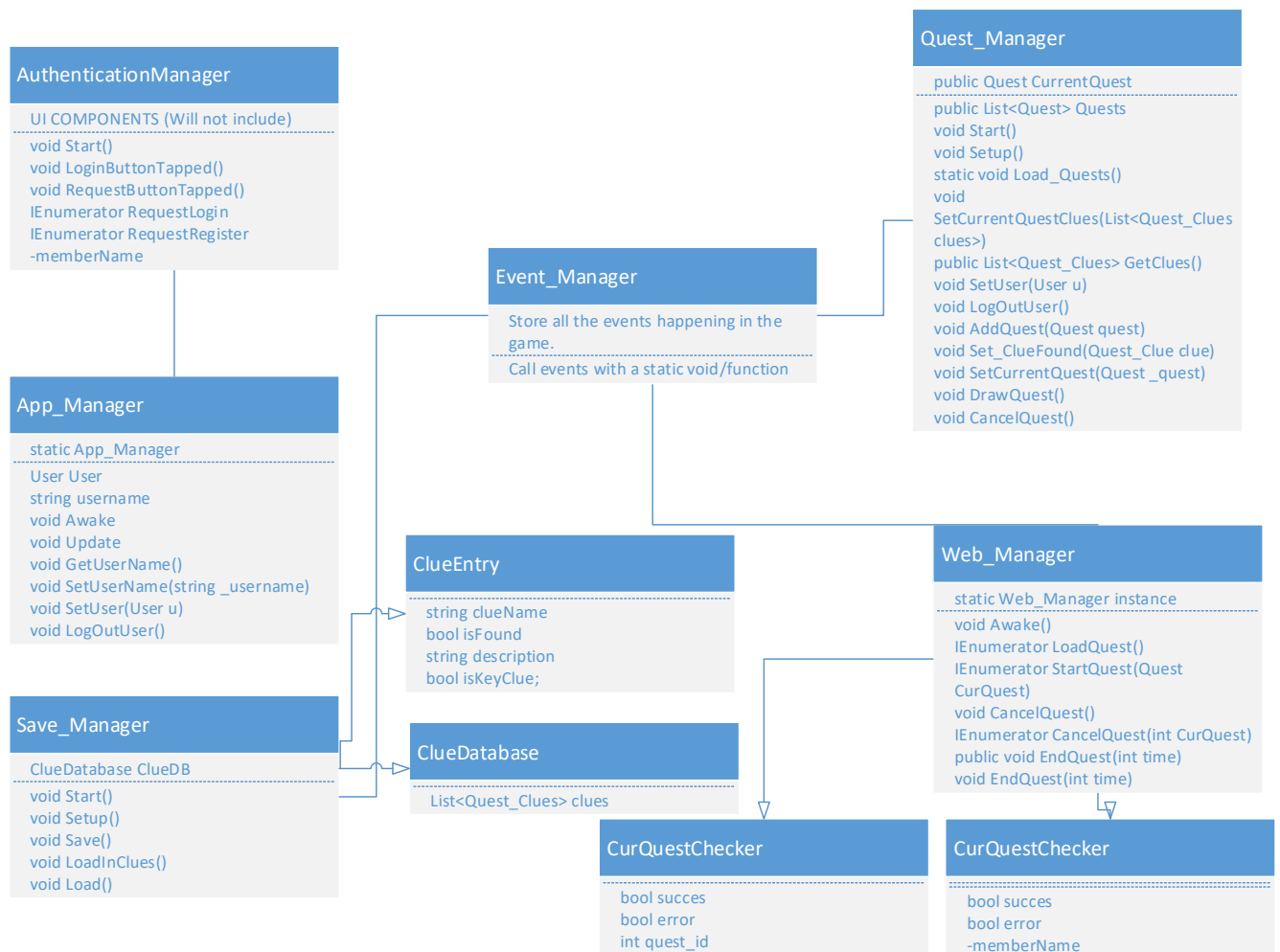
Naam: Sergi van Ravenswaay  
Studentnummer: 3017869

## Mijn tool

De tool genaamd DTABank is gemaakt om het werk van de designers en/of artists makkelijker te maken. Deze kunnen via de tool de data in de game aanpassen zonder dat zij in Unity hoeven te werken. Naast dat hun daar niet in hoeven te werken, scheelt het de developers ook tijd en kunnen zij verder gaan met developen. De tool zorgt er ook voor dat er niet elke keer een nieuwe build hoeft te worden gemaakt. Dit kan je gewoon via de website doen.

## Editorwindow

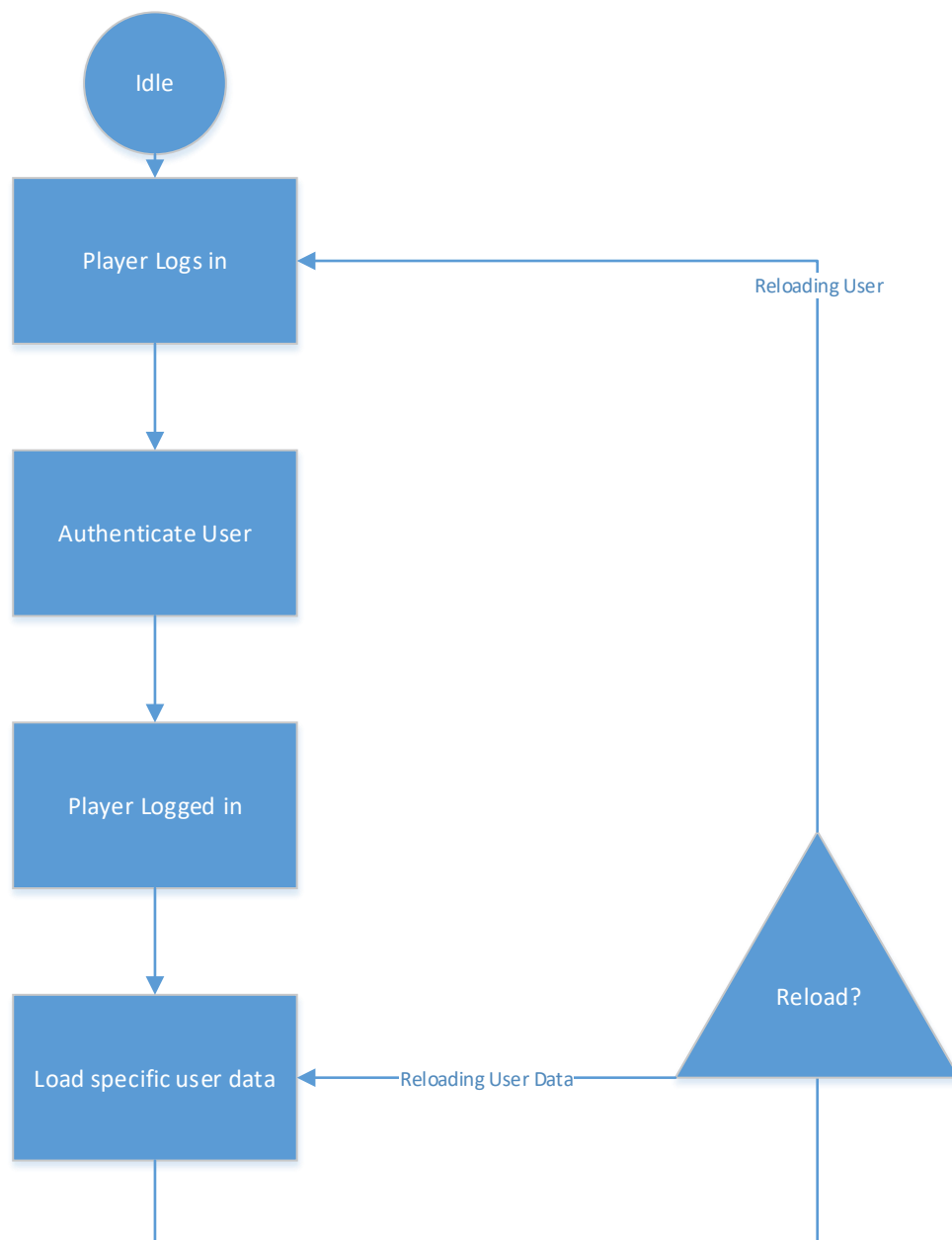
Er is in deze versie van de tool al een editorwindow. Deze is nog niet uitgebreid, maar je kan in Playmodus wel de quests refreshen en een overview krijgen van de quests!



## UML

Hieronder vind je de UML van de script in de Engine. De reden dat er zo weinig scripts zijn, is omdat het meeste managen van het werk buiten de engine gebeurt. Binnen de engine word er alleen maar data opgevraagd, gecontroleerd en gebruikt.

De event\_manager is in mijn code waar alles samenkomt. Vanuit deze singleton kan je alles in de code aanroepen.



### Activity Diagram

Wat er voor nu achter de schermen gebeurt bij Unity is dat je kan inloggen en registreren. Omdat het Registreren hetzelfde werkt als het inloggen heb ik hier het inloggen getoont. Na het inloggen zal hij user specifieke en globale data uit de database ophalen. Deze kan je reloaden, maar je kan ook uitloggen en de hele user reloaden.

## Design Patterns

Factory Pattern – Om alle data in de tool goed te organiseren.

Singleton – Om met events snel bij andere data te kunnen en de webmanager vanaf elk moment te kunnen aanroepen.

Observer Pattern – Als een user klikt op een knop dan gebeuren er events

## Waarom heb je deze tool zo gemaakt?

Deze tool speelt zich met een reden een groot deel af buiten de Engine. Dit zodat hij op zoveel mogelijk platformen (hij draait nu op een website) werkt en dat je hem gemakkelijk kan aanpassen. Ook word de database via een MYSQL database beheerd. Omdat MYSQL zo goed samenwerkt met het website gedeelte. Kan je hier snel data aan toevoegen en deze via een JSON ophalen uit de database.

## Scripts

AuthenticationManager – Het inloggedeelte van de applicatie regelen. Zo heeft elke user zijn eigen gegevens

App\_Manager – Die alle data van de applicatie beheert en de Username etc kan opslaan voor als je van scene wisselt.

Web-Manager – Deze regelt alles met de server. Zo kan je vanaf daar alle website communicerende functies bouwen.

Quest-Manager – De data in dit geval de quests worden vanuit hier beheerd

Quest – Dit is een van de items die word opgeslagen met JSON en geladen vanuit JSON.

Save-Manager – De SaveManager kan de classes (Quest of Clues) opslaan en inladen. Zo kan hij bijvoorbeeld checken of jij al een quest bent gestart.

Event\_Manager – Hieruit worden alle events uitgeroepen en kan je makkelijk vanuit de hele game quests uitroepen. Deze staan ook overzichtelijk op een plek.

## Hoe is het ontwerp in het loop van het proces verandert?

Het ontwerp is redelijk hetzelfde gebleven. Vooraf was mijn plan dit te maken voor Hybrid Space. Dit zodat mijn designers de tool konden gebruiken en ik niet veel werk hoefde te doen. Het data opslaan en uitladen ging makkelijk

### *String splitten*

Eerst gebruikte ik een manier van string splitten. Ik laadde een string vanuit een webpagina en splitte deze door verschillende tekens uit elkaar. Dit was super sloom en werkte niet voor mij. Vandaar dat ik deze heb aangepast en heb gezorgd dat ik een JSON array terug krijg. Deze kan ik dan makkelijk met een scriptje omzetten naar een C# en die zorgt ervoor dat ik deze gemakkelijk in C# kan gebruiken. Naast stukken sneller scheelde het ook een hoop code!

## **Waar ik niet aan toegekomen ben**

Ik ben voor mijzelf niet toe gekomen aan het maken van een goede versie om jullie te laten zien wat de tool wel doet. Niet in de browser, maar in Unity. Er zijn nog veel meer mogelijkheden buiten de webbrowser en ik zou Unity echt zo kunnen bouwen dat het heel handig te gebruiken is!

## **Wat zou je de volgende keer willen toevoegen?**

De volgende keer zou ik graag meer in de editor ook nog van die kleine componenten hebben. Deze kunnen er dan voor zorgen dat je niet de hele tijd naar de website hoeft te switchen. Zo wil ik proberen nog een tooltje te maken dat je database makkelijk kan beheren vanuit Unity. Dit zou een hele hoop werk schelen en het testen (ook buiten playmodus) veel sneller maken!

## **Conclusie**

Mijn conclusie is dat het concept goed is uitgewerkt. Deze tool ga ik zeker verder gebruiken voor usercontrol en data management. Ik zou deze tool zeker willen uitbreiden en verder willen integreren in de Unity Editor.

## **Data Sten Duindam en Conventions**

### *Data Sten Duindam*

De data van Sten Duindam vond ik niet handig om via mijn tool in de editor om te zetten. Daarom heb ik een extra kopje op het DTABoard dashboard gemaakt waar je handig blokken kan toevoegen. Zo kan je zijn data nog makkelijk toevoegen.

### *Save Functie*

De save/laad functie word in dit project nog niet gebruikt. Dit komt omdat het niet handig was om hier in te bouwen aangezien ik dit heb moeten lostrekken van mijn Hybrid Space project. Mocht je de save functie in werking zien, dan zou ik op mijn git bij de repostery 'HS' kijken.

## **Belangrijke info**

Github HS:

<https://github.com/S3R9I/UIB>

Website:

<http://81.169.177.181/UIB/dashboard/>

Login:

docent

Wachtwoord:

test