Live Performance

Stephen Fiers

Klas: S23

Ontwerpdocument

Studentnummer: 2219611

Inhoud

[Versiebeheer 3](#_Toc485649437)

[Databaseontwerp 4](#_Toc485649438)

[Uitleg 4](#_Toc485649439)

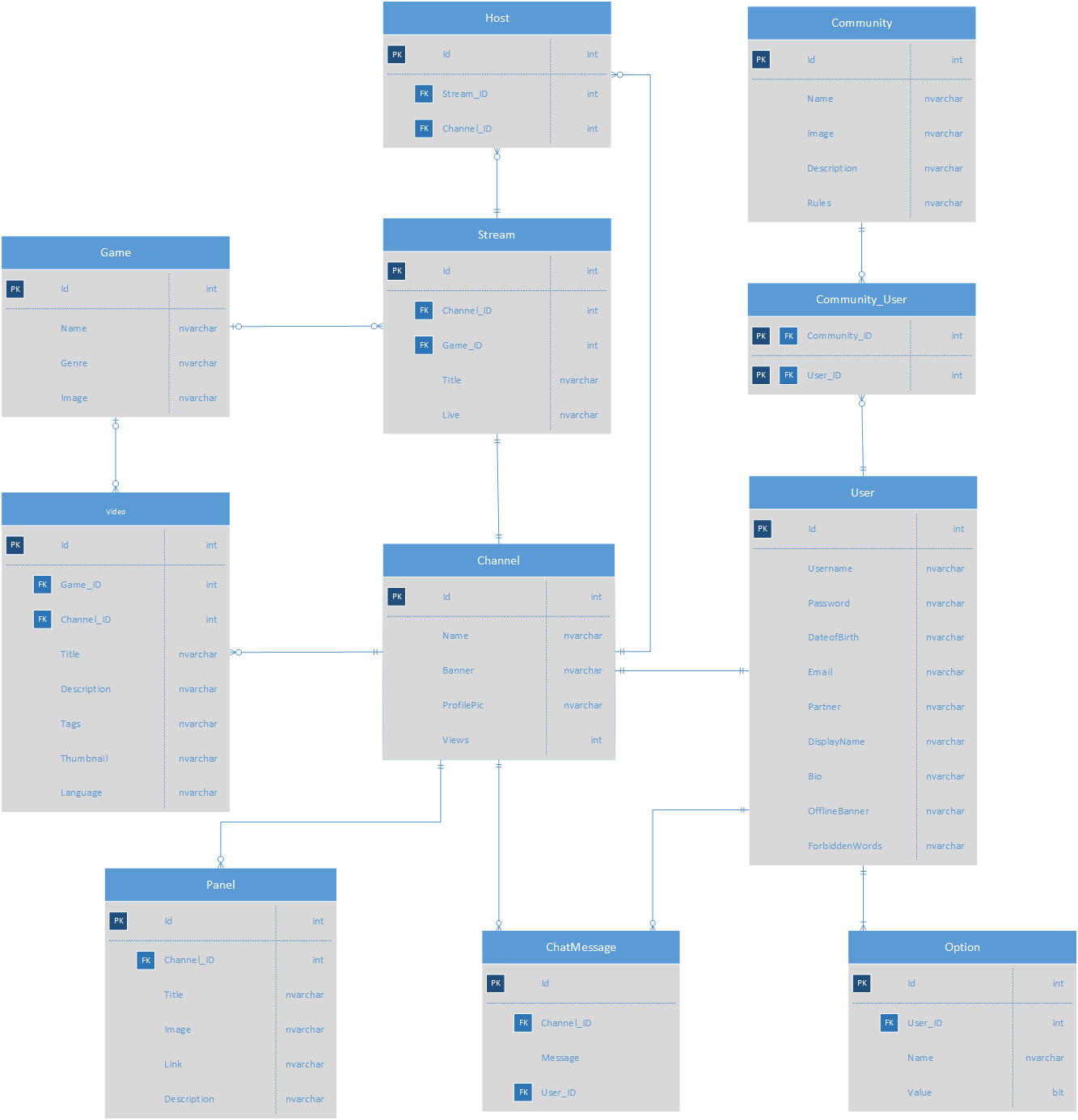
[Klassendiagram 6](#_Toc485649440)

[Uitleg 7](#_Toc485649441)

## Versiebeheer

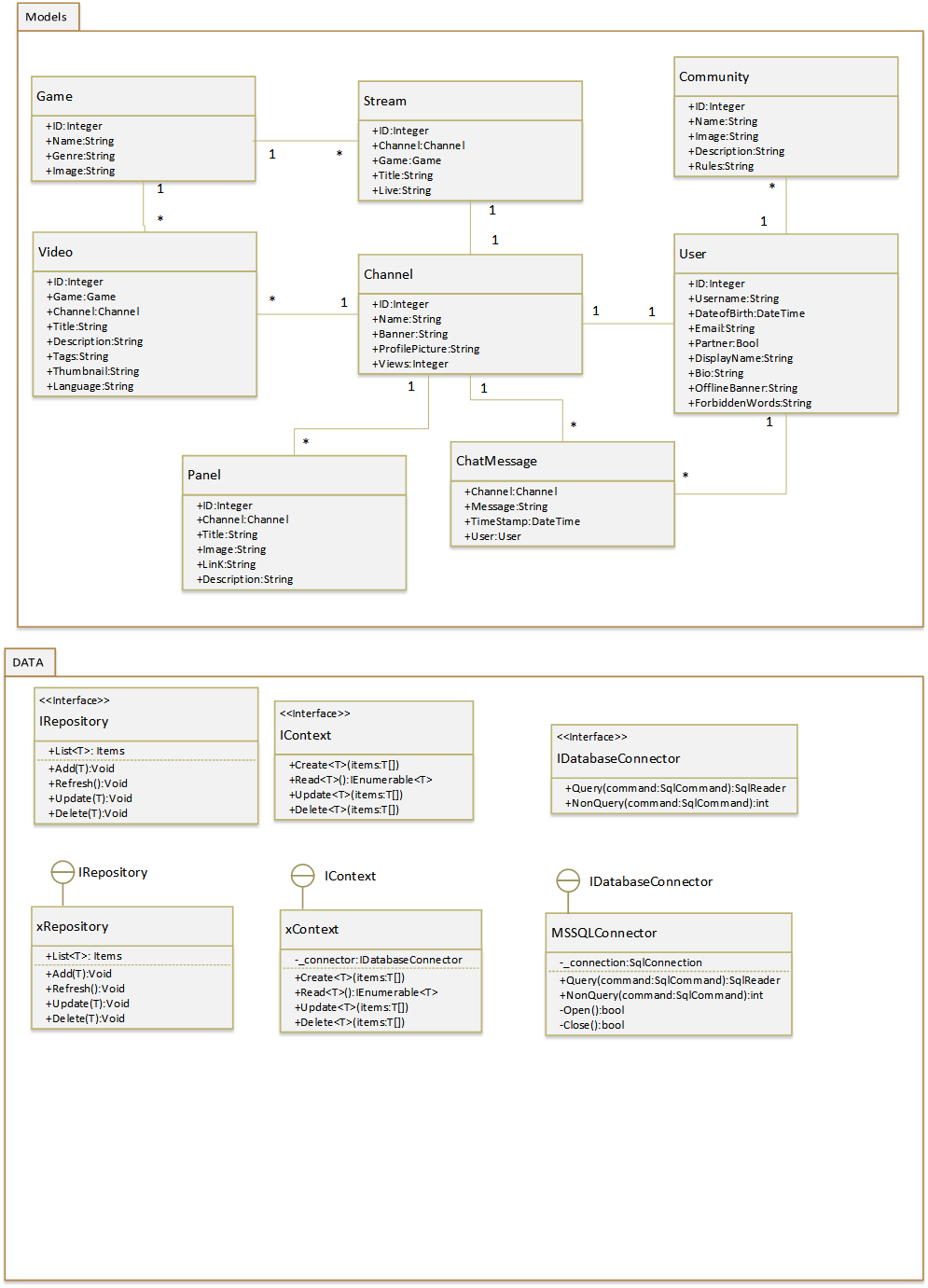
|  |  |  |
| --- | --- | --- |
| Versie | Aanpassingen | Tijd – Datum |
| V0.1 | Mock data in document gezet voor soepele start LP | 13.30 – 19/6/17 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Databaseontwerp



Uitleg  
Het systeem is gebaseerd op gebruikers die ervaringen met elkaar delen. Vandaar dat de User ook het centrale punt van het programma is samen met zijn Channel. Channel en User zijn 1 op 1 gekoppeld met elkaars ID’s. Op een Channel kun je een Stream zien, de reden dat deze gescheiden zijn is omdat een persoon met zijn Channel een ander persoon z’n Stream kan hosten op zijn kanaal. Wat ook een leuke functie is, is het delen van Video’s op je kanaal om, als je niet live bent, een idee te geven waar jouw Channel om draait. Op een Channel kan de User ook Panelen maken. Deze geven informatie over het kanaal voor tijdens het kijken en voor als het kanaal offline is. Users kunnen bij een Community horen, zo kunnen zij aan anderen laten zien waar zij fan van zijn of bij horen.

## Klassendiagram



### Uitleg

Bovenin heb ik mijn models uitgewerkt en daarin aangegeven welke relaties models met elkaar hebben. In het Data gedeelte kun je zien dat er gebruik wordt gemaakt van een Interface voor de database connectie, deze wordt gebruikt door de MSSQLConnector die op zijn beurt weer verbinding maakt met de server. Ook heb ik voor Repositories en Contexten een interface gemaakt, omdat ze zich op deze manier aan dezelfde regels houden. De x bij xRepository en xContext staan voor Type. Hier kan bijvoorbeeld User ingevuld worden of ChatMessage.