

Risicoanalyse

MMORPG ARCHITECTUUR – 2D TOPDOWN MEDIEVAL

Finn Alberts, Aaron Dang, Laurent Dassen en
Peter Derks
ZUYD HOGESCHOOL | HBO ICT

**ZU
YD**

Inhoud

1 Risicoanalyse	3
2 Maatregelen.....	4
2.1 Performance, scalability and elasticity van de game world.....	4
2.2 Elasticity van de character management en social systems	5
2.3 Performance en security van de game client	5

1 Risicoanalyse

In Tabel 1 is de risicoanalyse te zien voor de verschillende componenten op basis van de verschillende architectuurkarakteristieken. Ieder karakteristiek wordt volgens Figuur 1 beoordeeld.

Voor de impact (severity) van een risico wordt de volgende beoordeling aangehouden:

- 1: de impact is verwaarloosbaar en/of heeft weinig invloed op de ervaring van gebruikers;
- 2: de impact is voelbaar voor gebruikers, maar wel acceptabel;
- 3: de impact bij de gebruikers is groot en vormt een risico dat gebruikers stoppen met spelen en/of het bedrijf (imago)schade oploopt.

Voor de kans (likelihood) van een risico wordt de volgende beoordeling aangehouden:

- 1: de kans is minimaal, uitgaande van de standaardmaatregelen die binnen de ICT worden aangehouden¹;
- 2: de kans is aanwezig, maar wel te overzien;
- 3: de kans is groot dat dit een risico vormt.

LIKELIHOOD ↓	SEVERITY →		
	1 Minor	2 Serious	3 Fatal
1 Unlikely	LOW - 1 -	LOW - 2 -	MEDIUM - 3 -
2 Possible	LOW - 2 -	MEDIUM - 4 -	HIGH - 6 -
3 Probable	MEDIUM - 3 -	HIGH - 6 -	HIGH - 9 -

Figuur 1 Beoordeling van karakteristieken

In Tabel 1 is deze beoordeling weergegeven met eerst de kans (likelihood), vervolgens een * en daarna de impact (severity). Daaronder staat de score.

¹ Met standaardmaatregelen worden maatregelen bedoeld die binnen de ICT standaard worden genomen. Een voorbeeld is bijvoorbeeld het hashen van wachtwoorden binnen de database.

Tabel 1 Risicoanalyse

	Account management	Character management	Game world	Load and distribute management	Game client	Game store	Market	Online payment	Social systems	Coordinator
Performance	1*1 1	2*2 4	3*3 9	3*1 3	2*3 6	1*1 1	1*2 2	1*2 2	1*2 2	1*1 1
Scalability	1*2 2	1*3 3	2*3 6	1*2 2	1*1 1	1*2 2	1*2 2	1*2 2	1*3 3	1*2 2
Elasticity	1*1 1	2*3 6	3*3 9	1*2 2	1*1 1	1*2 2	1*2 2	1*1 1	2*3 6	1*2 2
Availability	1*2 2	1*3 3	1*3 3	1*3 3	1*1 1	1*2 2	1*2 2	2*2 4	1*2 2	1*3 3
Fault tolerance	1*3 3	1*2 2	2*2 4	1*1 1	2*1 2	1*3 3	1*3 3	1*3 3	1*1 1	1*1 1
Security	1*3 3	1*2 2	2*2 4	1*1 1	3*2 6	2*1 2	2*2 4	1*3 3	1*2 2	1*1 1
Data integrity	1*3 3	2*2 4	1*2 2	1*1 1	1*1 1	1*2 2	1*3 3	1*3 3	1*2 2	1*1 1
Maintainability	1*1 1	2*2 4	2*2 4	1*2 2	1*2 2	1*2 2	1*1 1	1*1 1	1*1 1	2*1 2
Recoverability	1*3 3	1*3 3	1*1 1	1*1 1	1*1 1	1*2 2	1*3 3	1*3 3	1*1 1	1*1 1

2 Maatregelen

Om de grootste risico's (rood gemarkeerd in Tabel 1) te beperken, worden maatregelen genomen. Daarnaast wordt bij het ontwerpen van de architectuur gekozen voor een architectuurstijl die sterk is in performance, scalability en elasticity. Dit zijn de drie architectuurkarakteristieken waar de meeste risico's in zitten. Voornamelijk de performance en elasticity van de game world zijn grote risico's.

2.1 Performance, scalability and elasticity van de game world

De game world moet vele spelers tegelijkertijd aankunnen. Ook als het aantal spelers steeds groter wordt en er 'spitsuren' ontstaan waarop het enorm druk is. Het aantal interacties met deze componenten kan dus enorm toenemen, waardoor deze componenten een grotere latency krijgen. Dit vormt een groot risico, omdat een grote latency zorgt voor meer lag. Lag kan de reputatie van de game aantasten en kan tevens (een deel van) de spelerbase ontmoedigen om te spelen.

Om de performance van de game world te waarborgen en te zorgen dat de game world scalable en elastisch is, wordt voor de communicatie tussen de game client en game world een event-driven architectuur gebruikt. Het load management component zorgt er hierbij voor dat de events gelijkmatig over de servers worden verdeeld. Hierbij is het uitgangspunt dat events die binnen dezelfde in-game locatie worden uitgevoerd zoveel mogelijk door dezelfde server worden

uitgevoerd. De reden hiervoor is dat acties binnen een locatie invloed (kunnen) hebben op andere acties op dezelfde locatie.

Om de belasting op een server verder te minimaliseren worden acties in beginsel binnen de game client uitgevoerd. Deze triggeren een event, waarbij de server controleert of de actie uitgevoerd kan en mag worden (dat wil zeggen: er wordt niet vals gespeeld middels een aangepaste game client). Indien een actie niet uitgevoerd kon/mocht worden, wordt deze actie teruggedraaid voor de game client. Dit vermindert tevens de latency die de game client ervaart.

2.2 Elasticity van de character management en social systems

Net als bij de game world kan een groeiend spelersaantal voor problemen zorgen binnen deze componenten. Dit vormt dezelfde risico's, waarbij lag ervoor zorgt dat de reputatie van de MMORPG kan worden aangetast en (een deel van) de spelersbase wordt ontmoedigd om te spelen. Lag is binnen dit component echter iets acceptabeler als binnen de game world, omdat (bijvoorbeeld) een extra seconde vertraging voordat een voorwerp in de inventaris verschijnt minder frustrerend is voor een gebruiker, dan wanneer een gebruiker niet goed kan rondlopen binnen de spelwereld. Daarnaast is de kans op dit risico meer te overzien binnen dit component, omdat er minder interacties met dit component zullen zijn in vergelijking met de game world. Desalniettemin zal dit een groot aantal interacties zijn.

Om te voorkomen dat de lag onacceptabel groot wordt, worden deze componenten schaalbaar gebouwd. Dat wil zeggen dat er meerdere servers kunnen worden gestart voor ieder van deze componenten.

2.3 Performance en security van de game client

De performance van de game client heeft direct invloed op de ervaring van de gebruiker en heeft daarmee grote impact. Een slechte ervaring kan ervoor zorgen dat een gebruiker besluit om niet verder te spelen. Daarnaast is de kans op dit risico groot, omdat de hardware (desktop/laptop) die verschillende gebruikers te beschikking hebben erg uiteen loopt. Deze kans is voor consoles kleiner.

Ook security vormt binnen de game client een risico, omdat de game client niet server-side draait. Dit maakt het mogelijk voor hackers om wijzigingen aan te brengen binnen dit component, om zo voor oneerlijke voordelen ten opzichte van andere spelers te zorgen.

Om de performance van de game client te waarborgen wordt voor een pixelart artstyle gekozen, waardoor de grafische weergave minder intensief is voor de game client en er meer rekenkracht beschikbaar blijft voor andere taken die door de game client moeten worden uitgevoerd. Daarnaast moet het ook mogelijk zijn om in een lagere resolutie te spelen.

De game client is het meest kwetsbare onderdeel van het systeem, omdat het voor gebruikers die vals willen spelen het makkelijkst aan te passen is. Om te voorkomen dat dit invloed heeft op andere spelers worden alle acties server-side geverifieerd en wordt kwetsbare informatie nooit via de game client van een component naar een ander component doorgevoerd. De enige uitzondering hierop is de inlogsessie. Deze wordt binnen de game client opgeslagen middels een unieke identifier, die tevens iedere sessie verandert.

Met opmerkingen [AF(1): TODO MAATREGELEN

Met opmerkingen [DL(2R1): hoi