# Requirements EasySockets

Hieronder staan de requirements van EasySocket beschreven. Het doel van dit document is om helder te krijgen wat EasySockets precies moet doen en waarom. De formulering gaat in de vorm van user stories in combinatie met acceptatiecriteria.

Het doel van EasySockets is om het ontwikkelen van websocket-gedreven applicaties voor ontwikkelaars makkelijker te maken. Dit geldt niet alleen voor applicaties met websockets als hoofd-component, maar ook voor applicaties die kleinere websocket-gedreven componenten bevatten.

## Rollen

1. Developer
2. Ontwikkelaar van EasySockets

Het is zo dat in sommige user stories wordt gerefereerd naar ‘cliënt’ of ‘cliënten’. Hiermee wordt niet verwezen naar de Developer of Ontwikkelaar van EasySockets. De cliënt kan een webgebruiker zijn die verbindt met de server, maar ook een microcontroller met internetverbinding, of een netwerkapplicatie zoals Postman. Het is aan de Developer om te bepalen wie/wat de cliënt is. EasySockets zorgt ervoor dat je dit zelf kan bepalen en aanpassen naar de behoeften van de Developer.

Dit is ook de reden dat er geen user stories te vinden zijn voor deze cliënten. Omdat een cliënt alles kan zijn (zolang het een websocket connectie kan maken), is het onmogelijk om vanuit EasySockets iets te ontwikkelen. Het is aan de Developer om zijn cliënt zo te ontwikkelen dat hij met de server kan communiceren.

## Developer

De developer is degene die de code van EasySockets (ook wel ‘source code’ genoemd) downloadt en gebruikt in zijn applicatie. De ontwikkelaar is de eindgebruiker van EasySockets.

De developer wilt met gemak krachtige en aangepaste websocket applicaties kunnen coderen. Hij wilt krachtige features om ervoor te zorgen dat hij zijn eigen protocollen kan schrijven. Dit bevat een eigen authenticatie en autorisatie pipeline en event driven development in combinatie met custom event binding. De websockets moeten gemanipuleerd kunnen worden buiten de websocket instance om.

### Koppelen van websocket-typen

De volgende user stories zijn gerelateerd aan het koppelen van websocket-types aan urls.

#### Als developer wil ik EasySocket-types kunnen koppelen aan een url zodat een websocket connectie verbonden kan worden met de server.

Acceptatiecriteria: 1. Voordat de IApplicationBuilder.Run methode wordt aangeroepen moet er geconfigureerd kunnen worden welk websocket-type van toepassing is op welke url. 2. EasySocket-types moeten herbruikt kunnen worden op verschillende urls. 3. Urls mogen niet meerdere keren genoemd worden. Als dit wel gebeurt, moet het systeem een exceptie gooien.

#### Als developer wil ik een encoding aan kunnen geven zodat de websocket berichten van andere talen/encodings kan ontvangen.

Acceptatiecriteria: 1. Per configuratie voor een EasySocket type moet er een encoding aangegeven kunnen worden. 2. De standaard encoding is UTF8.

### Authenticatie en autorisatie

De volgende user stories zijn gerelateerd aan het authenticatie en autorisatie proces van EasySockets.

#### Als developer wil ik authenticatie zelf beheren zodat ik controle heb wie met de server verbonden mag worden.

Acceptatiecriteria: 1. De developer moet functionaliteit kunnen coderen om de cliënt to authenticeren. Dit wordt gedaan door middel van een boolean waarde. 2. Als de teruggestuurde waarde false is, zal de pipeline beëindigd worden en de request in een 401 statuscode resulteren. 3. Als de teruggestuurde waarde true is, zal het volgende item in de pipeline uitgevoerd worden. 4. Als er geen volgend item in de pipeline is, zal de websocket geaccepteerd worden.

#### Als developer wil ik autorisatie zelf beheren zodat ik controle heb over wie de cliënt is volgens het systeem.

Acceptatiecriteria: 1. De developer moet een unieke (string)waarde kunnen specificeren tijdens het autorisatieproces. 2. Als de developer geen waarde specificeert, wordt een standaard procedure gevolgd om de connectie alsnog een unieke waarde te geven.

#### Als developer wil ik dat cliënten opgedeeld worden in kamers zodat niet iedereen gedeelde informatie kan ontvangen.

Acceptatiecriteria: 1. Berichten worden die vanuit de server naar cliënten worden gestuurd, worden alleen ontvangen door cliënten in dezelfde kamer.

#### Als developer wil ik beheren welke cliënten in welke kamers opgedeeld worden zodat ik kan bepalen welke informatie naar welke cliënten gaat.

Acceptatiecriteria: 1. De developer moet kunnen specificeren in welke kamer de cliënt terechtkomt tijdens het autorisatieproces. 2. Als de developer geen waarde specificeert wordt een standaard procedure gevolgd om de connectie alsnog in een kamer te zetten.

#### Als developer wil ik dat ik authenticatie en autorisatie processen kan herbruiken zodat ik minder dubbele code hoef te schrijven.

Acceptatiecriteria: 1. authenticatie functionaliteit die de developer codeert moet herbruikt kunnen worden. 2. autorisatie functionaliteit die de developer codeert moet herbruikt kunnen worden.

#### Als developer wil ik de standaardwaardes voor authenticatie aan kunnen passen zodat ik minder dubbele code hoef te schrijven.

Acceptatiecriteria: 1. Als er geen authenticatieproces is gekoppeld aan een EasySocket-type, moet het systeem een standaardwaarde pakken die de cliënt autoriseert. 2. De developer moet deze standaardwaarde globaal kunnen configureren. 3. De developer moet deze standaardwaarde per EasySocket-type kunnen configureren.

#### Als developer wil ik de standaardwaardes voor autorisatie aan kunnen passen zodat ik minder dubbele code hoef te schrijven.

1. Als er na autorisatie geen unieke waarde is gekoppeld aan een connectie, moet het systeem een standaardprocedure volgen die de unieke waarde teruggeeft.
2. De developer moet dit proces zelf kunnen coderen naar zijn behoeften.
3. Als er na autorisatie geen kamer is gekoppeld aan de connectie van een cliënt moet het systeem een standaardprocedure volgen om de connectie in een kamer te krijgen.
4. De developer moet dit proces zelf kunnen coderen naar zijn behoeften.

#### Als developer wil ik mijn eigen services kunnen gebruiken in het authenticatieproces zodat ik mijn eigen authenticatieproces kan maken.

Acceptatiecriteria: 1. Er moet dependency injection (DI) mogelijk zijn in de authenticatieprocessen.

#### Als developer wil ik mijn eigen services kunnen gebruiken in het autorisatieproces zodat ik mijn eigen autorisatieproces kan maken.

Acceptatiecriteria: 1. Er moet DI mogelijk zijn in de autorisatieprocessen.

### Berichten ontvangen en sturen

De volgende user stories zijn gerelateerd aan het (gemakkelijk) ontvangen en versturen van berichten door middel van de websocket-connectie van de cliënt.

#### Als developer wil ik genotificeerd worden van een bericht als de cliënt er een stuurt zodat ik dat bericht kan verwerken.

Acceptatiecriteria: 1. De developer wordt binnen het EasySocket-type genotificeerd van het bericht als de cliënt er een stuurt. 2. De developer ontvangt het bericht als hij genotificeerd wordt.

#### Als developer wil ik eigen berichten kunnen sturen naar de cliënt zodat de cliënt genotificeerd wordt van het bericht.

Acceptatiecriteria: 1. De developer moet binnen het EasySocket-type een methode aan kunnen roepen die een bericht van de server naar de cliënt stuurt. 2. De developer moet buiten het EasySocket-type een methode aan kunnen roepen die een bericht van de server naar de gespecificeerde cliënt stuurt.

#### Als developer wil ik de grootte van de berichten zelf instellen zodat ik meer controle heb over performance.

Acceptatiecriteria: 1. De developer moet tijdens het configureren van ieder websocket-type de grootte van de chunk formaat kunnen specificeren. 2. Berichten worden ontvangen aan de hand van de grootte van het chunk formaat. Als berichten groter zijn dan het formaat, wordt de nieuwe informatie opnieuw geprobeerd ontvangen te worden totdat het volledige bericht ontvangen is.

#### Als developer wil ik kunnen coderen welke berichten naar welke kamer gestuurd worden zodat ik berichten naar cliënten binnen en buiten de cliënt zijn kamer kan sturen.

Acceptatiecriteria: 1. De developer moet binnen het EasySocket-type een methode aan kunnen roepen die een bericht stuurt naar alle cliënten binnen de cliënt zijn kamer. 3. De developer kan niet binnen het EasySocket-type een methode aanroepen die een bericht stuurt naar cliënten van een specifieke kamer. 4. De developer kan buiten het EasySocket-type een methode aanroepen die een bericht stuurt naar cliënten van een specifieke kamer. 5. De developer kan buiten het EasySocket-type een methode aanroepen die een bericht stuurt naar specifieke cliënten.

#### Als developer wil ik kunnen coderen welke berichten naar alle gebruikers gestuurd worden zodat ik alle cliënten op de hoogte kan houden.

Acceptatiecriteria: 1. De developer kan buiten het EasySocket-type een methode aanroepen die een bericht stuurt naar alle cliënten. 2. De developer kan binnen het EasySocket-type een methode aanroepen die een bericht stuurt naar alle cliënten buiten de cliënt zijn kamer.

### Event driven development

De volgende user stories zijn gerelateerd aan het (gemakkelijk) verwerken en opdelen van de code van een developer.

#### Als developer wil ik namen van evenement-types kunnen registreren zodat ik code kan aanroepen op basis van het evenement.

Acceptatiecriteria: 1. De developer moet zijn eigen code kunnen linken aan de naam van een evenement. 2. Het systeem registreert de namen en de achterliggende code. 3. Als een bericht ontvangen wordt waarvan de naam van het evenement gelijk is aan een van de aangegeven namen, moet de gelinkte code uitgevoerd worden.

#### Als developer wil ik evenement-types kunnen registreren zodat ik zelf kan bepalen hoe een evenement eruit ziet.

Acceptatiecriteria: 1. De developer moet een type kunnen registreren die als evenement dient. 2. De developer moet methodes implementeren die aangeven wat de naam en het bericht van het evenement is. 3. Als de cliënt een bericht verstuurd, moet het systeem de naam en bericht van het evenement extraheren en de bijhorende code aanroepen.

#### Als developer wil ik dat ik een melding krijg van een ongeregistreerd evenementen-type zodat ik sneller fouten kan opsporen.

Acceptatiecriteria: 1. De developer moet genotificeerd worden als er een naam van een evenement geregistreerd wordt dat de developer niet heeft gespecificeerd.

#### Als developer wil ik een melding krijgen van een fout tijdens het bepalen van evenementen-types zodat ik sneller fouten kan opsporen.

Acceptatiecriteria: 1. De developer moet genotificeerd worden als er een fout optreed tijdens het registreren van een evenement.

### Verwijderen websockets

De volgende user stories zijn gerelateerd aan het afsluiten van de connecties.

##### Als developer wil ik een melding ontvangen de connectie met een cliënt afsluit zodat ik dit kan verwerken.

Acceptatiecriteria: 1. De developer wordt genotificeerd als de connectie gesloten wordt, op welke manier ook.

#### Als developer wil ik de connectie af kunnen sluiten zodat de verbinding tussen de cliënt en de server wegvalt.

Acceptatiecriteria: 1. De developer kan binnen het EasySocket-type de connectie met de cliënt afsluiten. 2. De developer kan buiten het EasySocket-type de connectie met een cliënt afsluiten.

#### Als developer wil ik de sluit-status-beschrijving van de websocket per connectie in kunnen stellen zodat ik de cliënt de reden kan laten weten van de sluiting van de connectie.

Acceptatiecriteria: 1. Tijdens het configureren van een EasySocket-type wil ik de mogelijkheid hebben om de sluitbeschrijving te configureren.

#### Als developer wil ik alle connecties binnen een kamer afsluiten zodat de verbinding van alle cliënten wegvalt.

Acceptatiecriteria: 1. De developer kan buiten het EasySocket-type de connectie met alle cliënten binnen een kamer afsluiten. 2. De developer kan buiten het EasySocket-type de connectie met alle cliënten afsluiten.

### Compatibility

Dit zijn de user stories die gerelateerd zijn aan de compatibiliteit van EasySockets. Dit betekent dat EasySockets niet alleen op de nieuwste versies moet werken, maar ook op oudere.

#### Als developer wil ik dat EasySocket werkt op alle .NET Core versies zodat ik het pakket op zoveel mogelijk plekken kan gebruiken.

## Ontwikkelaar van EasySockets

De ontwikkelaar van EasySockets (vanaf nu ‘ontwikkelaar’) is degene die de achterliggende code van EasySockets beheert.

De ontwikkelaar wilt dat het pakket zo aantrekkelijk mogelijk is. Dit betekent dat de code werkt en veilig is. Andere ontwikkelaars moeten zonder zorgen de code kunnen gebruiken. Ook moet de code ‘snel’ zijn. Dit betekent dat er een minimale hoeveelheid geheugen moet worden ingenomen door het EasySockets pakket.

De ontwikkelaar van EasySockets wilt dat oude code blijft werken, ook als er later veranderingen worden gedaan. Als dit niet gebeurt noemen we dit een ‘Breaking Change’.

### Breaking Changes voorkomen

#### Als ontwikkelaar wil ik dat na een update van EasySockets oudere code blijft werken zodat developers geen veranderingen in hun code hoeven te maken na een update.

Acceptatiecriteria: 1. Na een toevoeging van een feature binnen EasySockets moeten oude functionaliteiten op dezelfde manier werken.

#### Als ontwikkelaar wil ik automatische tests zodat ik de code na een update niet handmatig hoef te testen.

Acceptatiecriteria: 1. De code wordt getest door middel van unit-tests.

##### Als ontwikkelaar wil ik genotificeerd worden als oudere code niet werkt na een update zodat ik de nieuwere code kan aanpassen.

Acceptatiecriteria: 1. Als er een update wordt uitgevoerd op het pakket, moeten er automatische checks gedaan worden die valideren of (oudere) code nog steeds werkt. 2. Dit wordt gedaan door middel van een Continuous Integration(CI) pipeline in github. 3. Als een van deze tests falen, moet de ontwikkelaar een notificatie ontvangen zodat hij deze fouten op kan lossen.

### Performance

Dit zijn alle user stories die gerelateerd zijn aan de snelheid van EasySockets. Als hier naar ‘snelheid’ wordt gerefereerd, bedoelen we daar minimale geheugengebruik mee.

In de praktijk is het gebruiken van geheugen de grootste snelheidskiller. Als er in de user stories iets staat als ‘ik wil dat EasySockets snel is’, bedoelen we daar mee ‘ik wil dat het EasySockets pakket zo min mogelijk geheugen gebruikt’.

#### Als ontwikkelaar wil ik dat EasySockets snel is zodat het pakket aantrekkelijker wordt voor andere developers.

#### Als ontwikkelaar wil ik kunnen testen hoe snel mijn code is zodat ik fouten kan voorkomen.

Acceptatiecriteria: 1. Er worden automatische tests uitgevoerd die letten op de hoeveelheid geheugen dat in gebruik is genomen. 2. Er wordt een waarschuwing gegeven als de hoeveelheid geheugen boven een afgesproken hoeveelheid is.

### Versiebeheer

#### Als ontwikkelaar wil ik de versie van mijn code kunnen beheren zodat ik fouten kan herstellen door de code terug te draaien.

Acceptatiecriteria: 1. Het versiebeheer wordt gedaan door middel van git.