[Date]

Reno Rovers

[Company name]

[Company address]

werkafspraken Geoprofs

[Document subtitle]

Contents

[Communicatie 2](#_Toc176351083)

[Afmelden 2](#_Toc176351084)

[Product owner 2](#_Toc176351085)

[Code Conventies 3](#_Toc176351086)

[Casings 3](#_Toc176351087)

[Commits 3](#_Toc176351088)

[Code Comments 3](#_Toc176351089)

# Communicatie

De communicatie binnen het team gaat via een groep op Discord.

## Afmelden

* Als je niet aanwezig kunt zijn communiceer je dit met de rest van het team. Laat weten waar je bent gebleven zodat, wanneer nodig, de rest hier mee door kan gaan.
* Meld je voor 9 uur af bij de rest van de groep.
* Mocht je al eerder weten dat je een dag niet aanwezig zal zijn laat dit dan zo spoedig mogelijk weten.

# Code Conventies

## Casings

Voor het schrijven van duidelijke en consistente code binnen ons project hanteren we specifieke conventies voor het gebruik van verschillende casings. Dit helpt niet alleen om de leesbaarheid van de code te bevorderen, maar ook om een uniforme stijl aan te houden die door alle teamleden wordt gevolgd.

**1. Pascal Casing (PascalCase):** PascalCasing wordt gebruikt voor de naamgeving van klassen en typen. Bij PascalCasing begint elk woord in de naam met een hoofdletter, zonder scheidingstekens tussen de woorden. Deze stijl zorgt ervoor dat klassen en typen duidelijk herkenbaar zijn in de code.

* **Voorbeeld:** CustomerOrder, InvoiceProcessor, UserProfile.

**2. Camel Casing (camelCase):** camelCasing wordt toegepast voor de naamgeving van variabelen, functies, en methoden. Bij camelCasing begint de naam met een kleine letter, terwijl elk volgend woord begint met een hoofdletter. Dit draagt bij aan een duidelijke identificatie van variabelen en functies binnen de code.

* **Voorbeeld:** orderAmount, calculateTotal, userProfile.

**3. Snake Casing (snake\_case):** snake\_case wordt gebruikt voor de naamgeving van variabelen en sleutels in JSON-bestanden. Deze stijl maakt gebruik van kleine letters waarbij woorden worden gescheiden door een underscore (\_). Dit zorgt voor consistentie en leesbaarheid in JSON-structuren en andere vergelijkbare contexten.

* **Voorbeeld:** order\_amount, customer\_name, product\_list.

Door deze conventies strikt toe te passen, zorgen we voor een eenduidige en professionele codebase die door alle teamleden makkelijk te begrijpen en te onderhouden is.

## Commits

Begin de commit-bericht met een werkwoord in de gebiedende wijs, bijvoorbeeld: "Add", "Fix", "Update", "Remove", “Refactor”. Hierop volgt een korte, duidelijke beschrijving van wat er gecommit is. Commits moeten in het Engels geschreven worden.

Aan het einde van de werkdag moet er een end-of-day commit gedaan worden.

## Code Comments

Gebruik op de juiste plekken comments om de code leesbaar te houden voor teamgenoten.

# SCRUM

## Daily standup

Aan het begin van de werkdag starten we met een daily standup. Hierin bespreken waar iedereen aan gaat werken die dag. De standup duurt niet langer dan 15 minuten.