

Analysedocument

Individueel Project - ordio

Robin van Hoof

2022

Inhoudsopgave

[Versiebeheer 2](#_Toc97799672)

[Inleiding 3](#_Toc97799673)

[Projectomschrijving 3](#_Toc97799674)

[Definities 4](#_Toc97799675)

[Gebruikers 5](#_Toc97799676)

[De restaurant-bezoeker 5](#_Toc97799677)

[De restauranteigenaar 5](#_Toc97799678)

[Het eetgezelschap 5](#_Toc97799679)

[Projecteisen 6](#_Toc97799680)

[Functional Requirements 6](#_Toc97799681)

[Guest 6](#_Toc97799682)

[Owner 6](#_Toc97799683)

[Non-Functional Requirements 6](#_Toc97799684)

# Versiebeheer

|  |  |  |
| --- | --- | --- |
| Versie | Datum | Aanpassing |
| v1.0 | 17-02-2022 | Opzet document |
| |  | | |  | | Voorpagina, Inhoudsopgave, Versiebeheer, Inleiding, Projectomschrijving, Definities, Projecteisen |

# Inleiding

Dit document zal de Analysefase voor mijn individueel project van Software semester 3, “**Ordio**”bevatten. In de toekomst zal naar dit project verwezen worden met de bovengenoemde naam.

Ordio zal de applicatie worden die ik ga ontwikkelen om de verschillende leerdoelen van dit semester aan te tonen. Deze leerdoelen zijn verder uitgewerkt in het document “Learning Outcome Reflection – Robin van Hoof.pdf” te vinden in de bijhorende GIT en inleverbox.

## Projectomschrijving

Om bij het begin te beginnen, wat zal Ordio worden en waarom heb ik gekozen voor dit project? Mijn visie van Ordio is een universeel platform dat restaurants kunnen gebruiken om tafels de mogelijkheid te geven via een tablet of mobiel apparaat snel en eenvoudig eten te bestellen. Ordio zal gericht zijn op tapas-stijl restaurants waarbij veel kleine gerechten in rondes besteld kunnen worden.

Ik ben op dit idee gekomen tijdens een diner in een restaurant waar ze een vergelijkbaar systeem gebruikte. Hier vielen mij veel fouten en onhandigheden op aan het betreffende systeem wat hier gebruikt werd.

Het voornaamste zicht van Ordio is dat het een universeel platform zal worden. Dit betekent dat het niet alleen voor gebruikers, een gast bij het restaurant, vanzelfsprekend moet werken en er goed uit moet zien, maar ook dat restauranteigenaars gemakkelijk hun menu-items in Ordio kunnen zetten. Dit betekent dat ook voor hen een vanzelfsprekende en goed uitziende front-end-interface gemaakt zal worden

# Definities

|  |  |
| --- | --- |
| Woord | Definitie |
| Ordio | De naam van het individueel project. |
| MVP | **M**inimum **V**iable **P**roduct, de minimale functionaliteit die de app moet bevatten om deze als “werkend” te zien |
| UI | **U**ser **I**nterface, de visuele aspecten van het front-end van de webpagina waar de gebruiker mee interacteert |
| UX | **U**ser E**x**perience, de gebruikerservaring die de gebruiker heeft tijdens het gebruiken van de applicatie |
| Gast/Guest | De restaurant-bezoeker-gebruiker van de applicatie. Deze term staat verder gedefinieerd in het hoofdstuk [“Gebruikers”](#_De_restaurant-bezoeker) |
| Eigenaar/Owner | De restaurant-eigenaar-gebruiker van de applicatie. Deze term staat verder gedefinieerd in het hoofdstuk [“Gebruikers”](#_De_restauranteigenaar) |
| Tafel/Table | De eetgezelschap-gebruiker van de applicatie. Deze term staat verder gedefinieerd in het hoofdstuk [“Gebruikers”](#_Het_eetgezelschap) |

# Gebruikers

Om een inzicht te krijgen wie belangrijke groepen zijn waarop gericht moet worden tijdens de ontwikkeling van het project is het belangrijk in zicht te krijgen wie de gebruikers van het product zullen worden. Het product heeft een aantal gebruikers:

## De restaurant-bezoeker

Een van de gebruiker, en dan meteen ook de belangrijkste gebruiker, zal de restaurant-bezoeker zijn. De belangrijkste eis voor deze groep zal een intuïtief en vanzelfsprekend bruikbare app zijn. Deze gebruiker geeft niet veel om hóe het werkt of wat erachter zit, als het maar snel en efficiënt werkt en er goed uit ziet. Naar deze gebruiker zal in de toekomst verwezen worden als “gast” of “guest”

## De restauranteigenaar

De tweede gebruiker van Ordio zal de restauranteigenaar zijn. Hun interesse zal vooral liggen in het gemakkelijk aan kunnen passen van het systeem voor hun restaurant. Dit zal betekenen dat voor hen een goede handig te gebruiken interface een vereiste nodig is die zij kunnen gebruiken om item van hun menukaart gemakkelijk toe te voegen. Naar deze gebruiker zal in de toekomst verwezen worden als “eigenaar” of “owner”

## Het eetgezelschap

Een derde groep gebruiker van Ordio lijkt erg op de restaurant-bezoeker: het eetgezelschap. Dit is een groep restaurantbezoekers die samen als één groep aan een tafel eten. De belangrijkste eis van deze groep zal een efficiënt te gebruiken app zijn zodat het gezelschap niet constant veel tijd hieraan hoeft te besteden, en meer tijd heeft het gezellig te hebben met het gezelschap. Naar deze gebruikersgroep zal in de toekomst verwezen worden als “tafel” or “table”

# Projecteisen

Om het project goed op start te brengen moeten eerst requirements en een Minimum Viable Product (MVP) opgesteld worden. Ik ga hiervoor aan de slag met MoSCoW en requirements in de vorm van User-Stories . Ik heb hiervoor gekozen omdat deze methoden, uit ervaring, erg effectief werken.

## Functional Requirements

### Guest

[M] G-01 A guest must be able to order items in rounds.

[M] L-G-01.2 A table can order an x amount of items per person per round.

[M] P-G-01.3 A guest should be able to remove added items again

[M] G-02 A guest should be able to filter through the menu for a specific category of items.

[M] G-03 A customer should be able to see added costs for items where this applies.

[M] P-G-03.2 A guest and owner must be able to see the racked up extra cost at any point.

[S] G-04 A guest should be able to add comments to ordered items for the kitchen to see.

[S] G-05 A round of items should only be actually ordered and sent to the kitchen after the guest has added their desired items and the round is sent through by the guest.

### Owner

[M] O-01 A restaurant owner must be able to easily add items to the digital menu through a separate manager application.

[M] P-O-01.2 An owner must be able to add extra costs to items.

[M] P-O-01.3 An owner must be able to add self-defined tags to items that can be filtered on.

[M] O-02 An owner must be able to set a maximum amount of items a table can order per person per round.

## Non-Functional Requirements

[M] NF-01 The frontend must be written in correct sensible English

[S] NF-02 The system should be able to send push-messages to subscribed systems when an order gets placed for a restaurant.

[S] P-NF-02.2 An owner should be able to subscribe, using an API call, an external system (for example receipt system) to receive push-messages

## Minimum Viable Product