Testplan

Auteurs: **Yavuz Babacan, Nemanja Jacimovic, Vakar Kemal, Yonah** Diallo

Klas: INF1D

Datum: 20 April 2022

**Inhoudsopgave**

[1 Inleiding 3](#_Toc70321768)

[1.1 Opdrachtformulering 3](#_Toc70321769)

[2 Testproducten Testplan 3](#_Toc70321770)

[3 Testaanpak 3](#_Toc70321771)

[3.1 Product Risico Analyse (PRA) 3](#_Toc70321772)

[3.2 Projectrisico’s 3](#_Toc70321773)

[3.3 Teststrategie 3](#_Toc70321774)

[3.4 Acceptatie 3](#_Toc70321775)

[4 Beheersing Testproces 4](#_Toc70321776)

[4.1 Rapportage bevindingen en issues 4](#_Toc70321777)

[4.2 Bevindingenregistratie 4](#_Toc70321778)

[4.3 Testgevallen 4](#_Toc70321779)

[5 Planning 4](#_Toc70321780)

# Inleiding

In dit testplan wordt beschreven hoe de console-applicatie voor een reserveringssysteem van Izjen Airlines wordt getest om zo de kwaliteit van de applicatie te waarborgen.

## Opdrachtformulering

*In bullits hieronder: Wat moet er getest worden? Wat wordt er niet getest? (indien van toepassing). Beschrijf de verschillende onderdelen van de applicatie (klant, medewerker en admin onderdeel bijv.)*

* Member-onderdeel
* Guest-onderdeel
* Admin-onderdeel
* Reserveringpagina (Stoel kiezen)
* Registratiepagina (Slaan de gegevens goed op?)
* Vluchtonderhoud (Worden de vluchten op de juiste wijze bewerkt?)
* Betaaloverzicht (Krijgt de gebruiker de juiste gegevens te zien?)

# Testproducten Testplan

*In bullits hieronder: welke deliverables?*

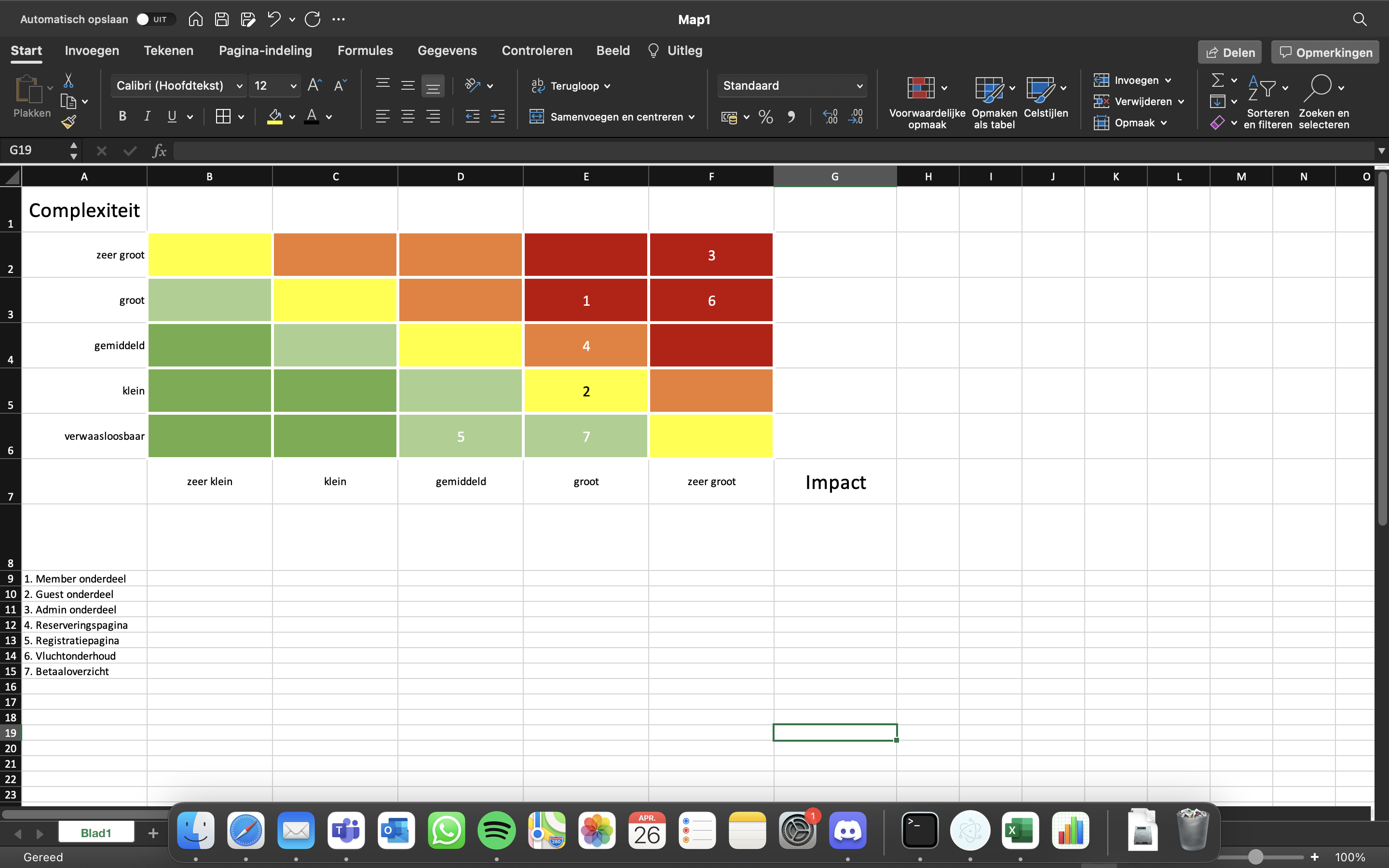
* Testresultaten
* Testgevallen
* Bevindingen

# Testaanpak

## Product Risico Analyse (PRA)

*Bepalen van de risico’s om de testinspanning en –gevallen te verdelen over de risico’s. Weergeven van het resultaat en / of verwijzen naar PRA in bijv. een Excel document of een visuele weergave.*

*Bij een PRA*



## Projectrisico’s

*Welke risico’s bestaan er rondom het project? Kunnen er mensen uitvallen vanwege ziekte bijvoorbeeld?*

* *Uitval wegens ziekte van een groepslid*
* *Uitval wegens motivatietekort*
* *Uitval wegens persoonlijke omstandigheden*
* *Tijdsnood*
* *Bestanden (Code) kwijtraken*
* *Product/Programma is onvoldoende getest*
* *Afwezigheid bij de les*
* *Gebrek aan communicatie*

## Teststrategie

*Hoe gaan wij testen oppakken? In welke sprint (indien nu al mogelijk om te bepalen…)?*

*Denk aan de verschillende testsoorten- en vormen en de restricties in tijd. Denk daarom ook aan mogelijkheden om tests gewoon samen te voegen. Geef dit wel hier aan. Laat deze aanpak goedkeuren! Hoe wordt de kwaliteit gewaarborgd?*

Per afgemaakte onderdeel wordt er een korte review/testmogelijkheid gehouden door de overige groepsleden. Hierin gaan wij kijken of de afgemaakte functie voldoet aan de eisen en in zijn geheel werkt zonder fouten. Op deze wijze kunnen wij de kwaliteit van ons programma waarborgen en weten wij zeker dat het voldoet aan de eisen van de Product Owner. Testsoorten die gebruikt zullen worden zijn:

Programmatest/Ontwikkeltest/Unittest:

Als testvorm voor de unittest maken we gebruik van een eenheidstest. Deze test wordt uitgevoerd tijdens de ontwikkelfase en test de functionaliteit van een apart onderdeel.

Systeemtest/Systeemintegratietest:

Bij het testen van het gehele systeem zullen we onder andere gebruik maken van regressietesten waarbij wij dus testen of alle onderdelen van het systeem nog werken na het maken van een wijziging.

Acceptatietest  
De testvorm die wij gebruiken voor onze acceptatietest is een usabilitytest. Hiermee kunnen wij de gebruiksvriendelijkheid van het eindproduct testen.

## Acceptatie

*Acceptatie van de applicatie. Voldoet het product aan de gestelde eisen? Hoe gaan wij het product tot acceptatie brengen? Wat zijn de afspraken met de opdrachtgever?*

We zetten onszelf in de voeten van een klant en lopen iedere functie die de gebruiker zou willen gebruiken door, hetzelfde doen we ook bij de admin (hierbij zullen we kijken of de admin alle benodigde functies kan benutten). Functies die niet werken of ontbreken zullen genoteerd worden en doorgegeven aan de developers. Als er volgens het ontwikkelteam iets onhaalbaar wordt zal er met de Product Owner worden overlegd voor een oplossing.

# Beheersing Testproces

## Rapportage bevindingen en issues

*Wat gaan wij hoe in welke frequentie rapporteren? Bevindingen / Testgevallen / status*

*Voortgang / impediments / Escalatie. Wanneer wel/ geen registratie?*

Tijdens het programmeren wordt de hele functie doorlopen om te kijken of het nog werkt zonder problemen en anders worden de problemen geregistreerd op het moment dat een bepaalde functie niet in zijn volledigheid werkt of het juiste aantoont. Dit om te voorkomen dat de gebruiker belemmerd wordt in zijn/haar gebruik van het programma. Ook worden bevindingen geregistreerd na gezamenlijke code reviews.

## Bevindingenregistratie

*Hoe / waar gaan wij de bevindingen vastleggen*

Alle bevindingen (zowel goed als fout) zullen worden vastgelegd in onze Trellobord. Hierin houden wij bij welke functies wel en niet werken en geven wij onder deze kopjes gedetailleerd weer wat er precies misgaat. Zo kunnen de overige groepsleden precies inzien wat er fout loopt en wellicht ondersteuning leveren waar nodig is.

## Testgevallen

*Hoe / waar gaan wij de testgevallen vastleggen*

De ontwikkelaars kunnen tijdens het zelfstandig werken hun nieuwe functies zelf testen. Bij het testen van de compatibiliteit tussen twee onderdelen van ontwikkelaars kunnen andere ontwikkelaars de functies testen, kijken of iets goed verloopt en of iets nog ontbreekt. Als we bij de voltooiing van het programma komen, kunnen we buitenstanders het programma laten doorlopen en noteren we de opmerkingen van de gebruiker.

# Planning

*Wie doet wat wanneer?*

Voor de planning wordt er gebruik gemaakt van een Trellobord. Hierin behouden wij onze taken en voortgang.

Link: <https://trello.com/b/iYu4gfDP/project-b>

