# Hogeschool Rotterdam / CMI

**Project B - INFANL03**

Opdracht: Documentatie testen

Vakar Kemal (1031225)

Yavuz Babacan (1035095)

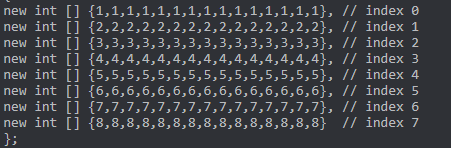
Yonah Diallo (1037538)

Nemanja Jacimovic (1022638)

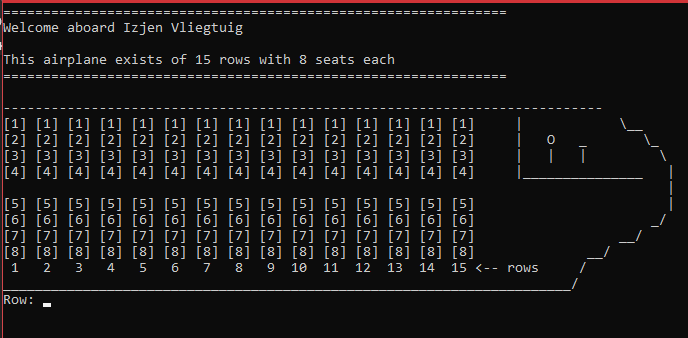
INF1D

1-6-2022

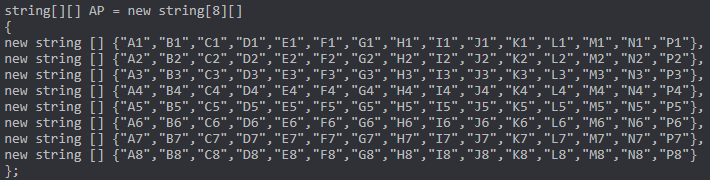
# Stoelen kiezen

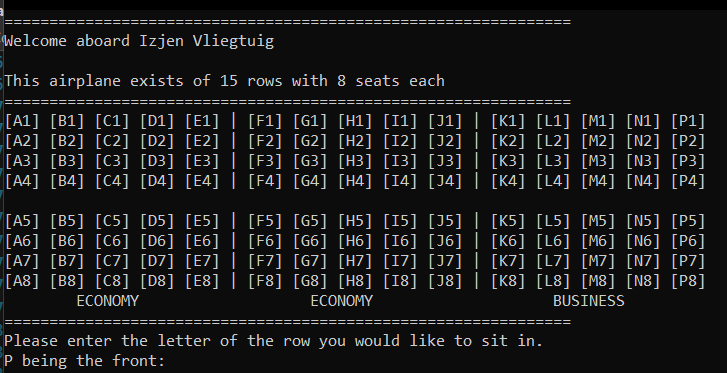


Een deel van de code van de eerste versie



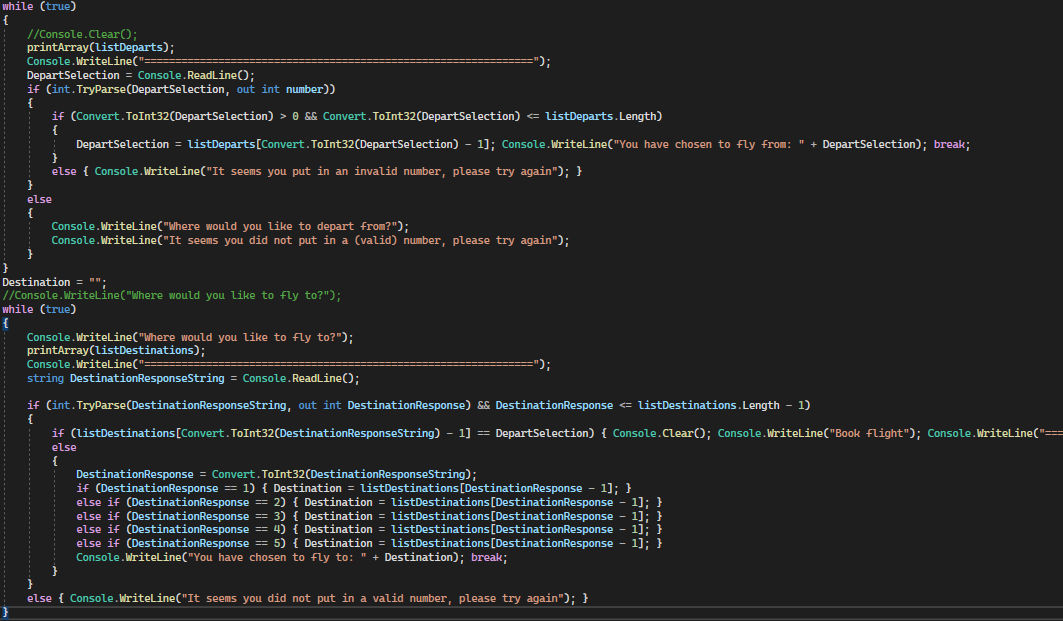
Zoals je hier kunt zien was dit de eerste versie van de stoelen reserveren, omdat het integers waren kon je in de array de uitgekozen stoel alleen veranderen naar een getal en dit was destijds 0. Dit werkte wel, maar het was onduidelijk om te zien welke stoelen nou precies gereserveerd waren. In de eerste versie waren er ook nog geen gordijnen die onderscheid maakten tussen de business en economy class.

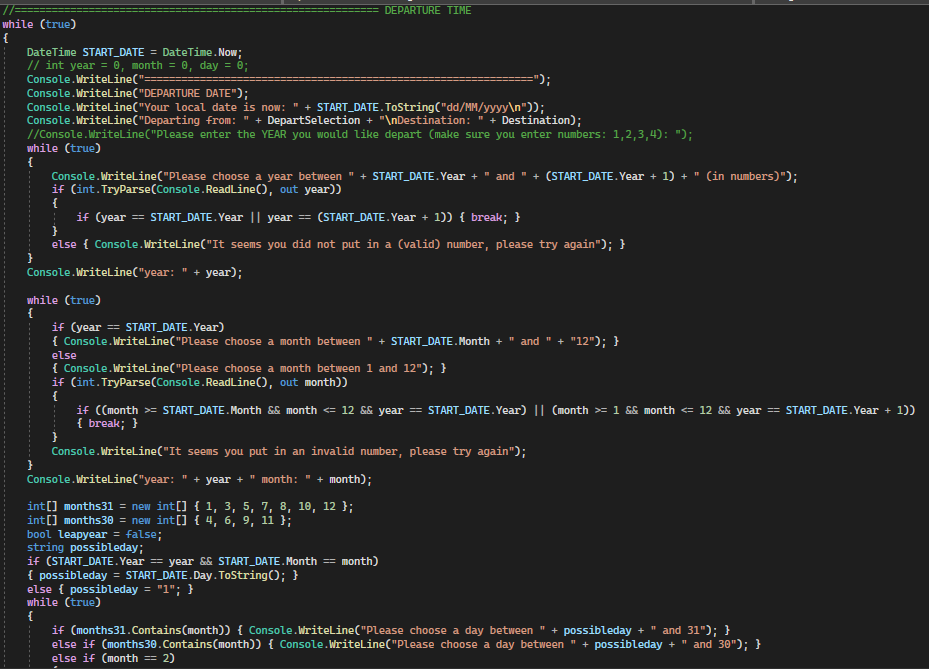


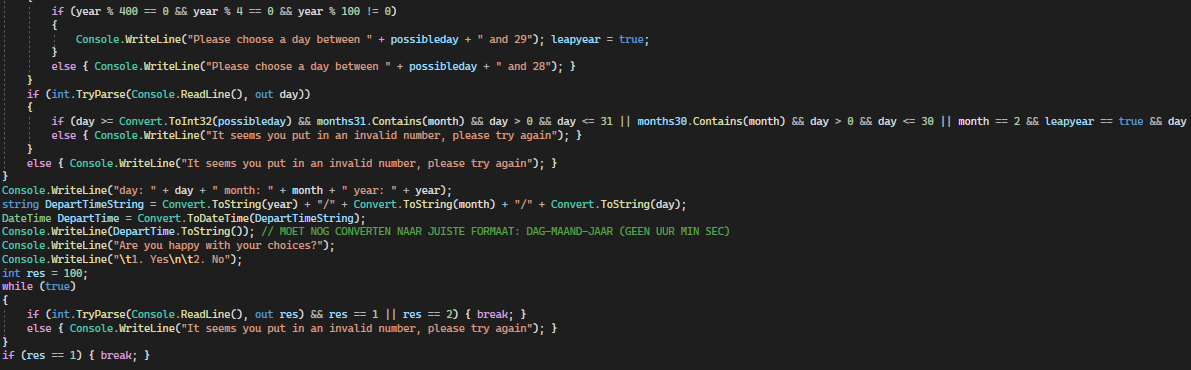


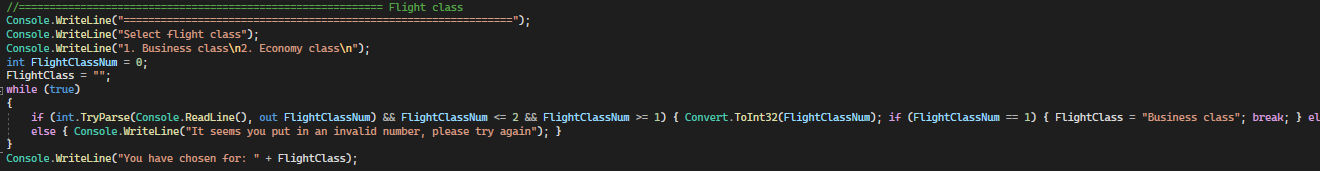
Hier de tweede versie. Ik had in plaats van een integer array nu een string array gebruikt zodat de gereserveerde stoelen duidelijker te zien waren. Nu kon ik ze veranderen naar een X als string en dit werkte heel goed. In de tweede versie is er ook een onderscheid aangegeven tussen de economy en businessclass.

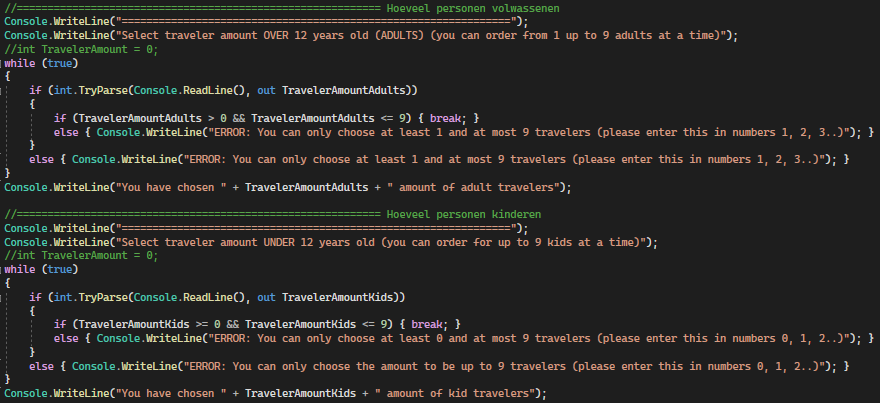
# Vlucht pagina

De vluchtpagina is veel veranderd, als aller eerst kon je alleen vertrek plaats en bestemming bepalen uit 1 lijst. Dat hebben we getest, en nu kan je de vertrek locatie en bestemming niet hetzelfde hebben. Op een of andere manier kon je ‘Londen’ niet kiezen, omdat deze aan het einde van het lijstje stond, dat hebben we gefikst.

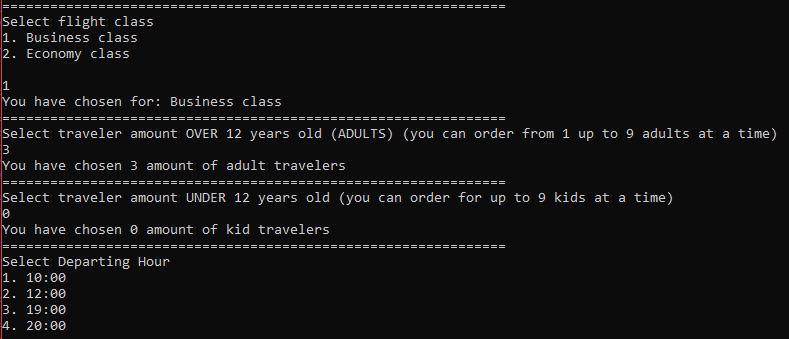
Daarna hadden we een ‘kiesdatum’ algoritme gemaakt. Dat ging niet zo goed, je kon in je datum naar het verleden kiezen, en toen kon je op de maanden van 30 dagen, 31 kiezen als dag. Dat is weggehaald, schrikkeljaar hebben we ook getest, februari heeft op schrikkeljaren 29 dagen en anders 28. Bij deze algoritme hebben we na testen een manier gevonden om de minimale invoerdatum up-to-date te houden zonder maintenance.

Daarna lieten we de gebruiker zijn/haar vlucht klasse kiezen tussen economy en business. Dit ging gelijk goed.



Dan hadden we een algoritme gemaakt waarbij de gebruiker kan kiezen hoeveel reizigers die wilde nemen. Eerst was dit beperkt tot 1, daarna werden het er 1 tot en met 9, en ten slotte kon de gebruiker aangeven hoeveel VOLWASSENEN en hoeveel KINDEREN. Eerst moest je mininmaal 1 kind meenemen, dat is dus veranderd naar minimaal 0.

Ten slotte kon je bij de vluchtpagina de vertrektijd in uur kon kiezen uit een lijst. Hierbij het testen ging het goed.



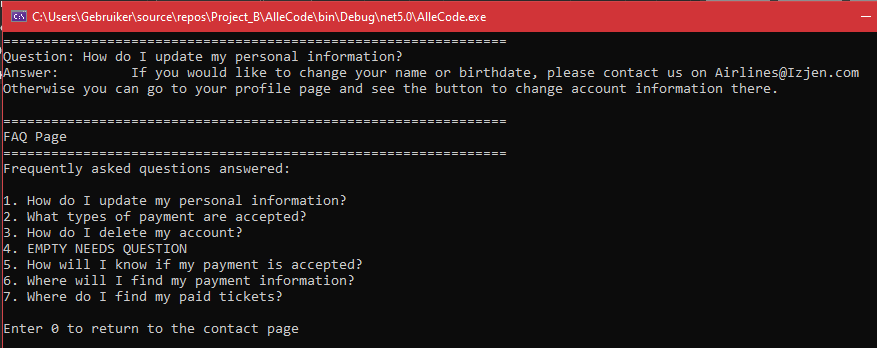
# Contact pagina

User story:  
Als reiziger wil ik een contact pagina zodat ik in contact kan komen met de luchtvaartmaatschappij.

Er was geen optie om terug te gaan van de contact pagina naar de menu pagina.

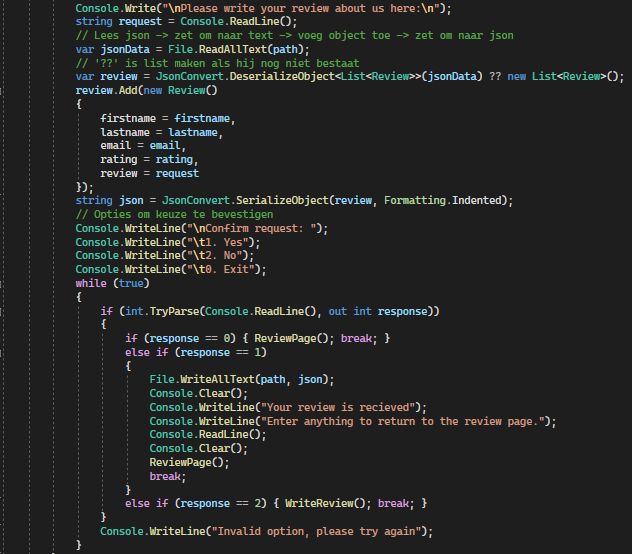
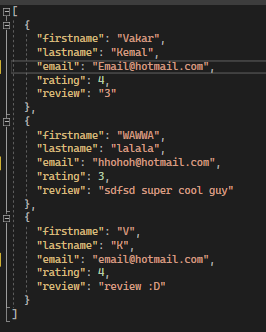
# FaQ pagina

User story:  
Als bezoeker wil ik een FAQ pagina zodat ik al mijn vragen beantwoord kan krijgen.

Eerst had je op 1 lijn, vraag en antwoord, nu kan je een vraag kiezen, en dan zie je bovenin het antwoord, en dit kan je oneindig vaak doen. Eerst kon dat niet. Daarnaast was het niet mogelijk om terug te gaan naar contact pagina, nu wel, en dit kan oneindig vaak. Vragen kunnen ook verwijderd en toegevoegd worden.

# Review pagina

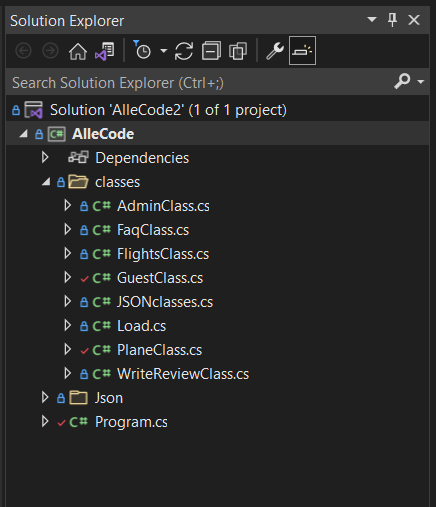
User story:  
Als gebruiker wil ik een Review pagina zodat ik mijn mening kan delen en die van anderen kan zien.

Eerst was het nog niet mogelijk om reviews terug te vinden, nu heb je een bestand hiervoor

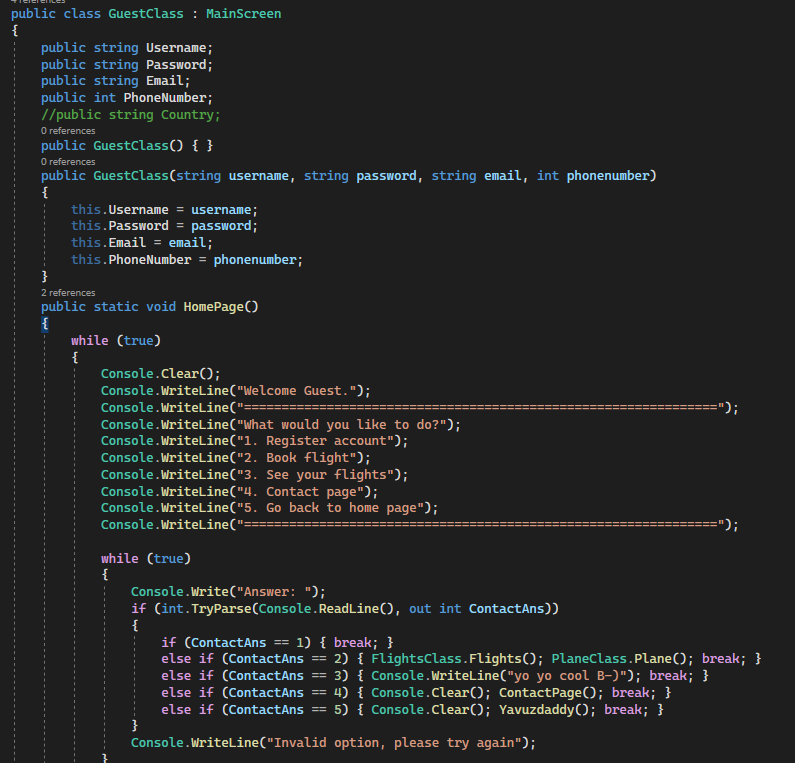
Je kon eerst de rating als getallen onder de 0 en boven de 5 invoeren of letters, nu niet meer  
De review met gegevens werden eerst niet opgeslagen, nu wel. Ook bij review page was het niet mogelijk om terug te gaan naar home page, hij crashte nadat de method klaar was met runnen. Dat is gefixt.



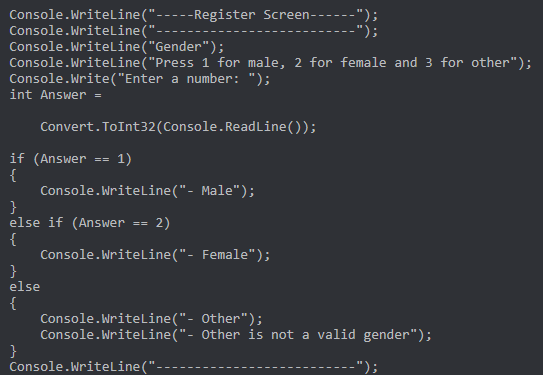
Zoals je hier kunt zien dit was onze oude code. Een spaghetticode. Hierin hadden we alles in de main staan en verdeeld in methods. Dit was zeer onhandig het werkte wel, maar het was zeer onduidelijk waar wat stond. De errors die we kregen waren moeilijk te fixen.



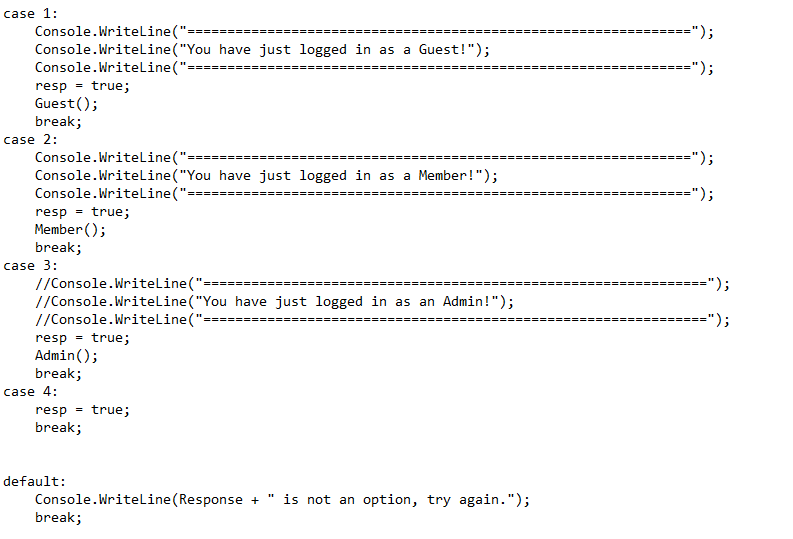
Hier kun je zien dat we alle methods hebben omgezet in classes. Hier staan ze allemaal op een rijtje. Dit werkte veel makkelijker en was veel overzichtelijker. Bugs fixen was zo een stuk makkelijker en de methods zijn nu veel makkelijker te vinden. Met feedback van de PO hebben we dit dus kunnen verbeteren. We zijn zeer tevreden over deze verbetering, omdat de code er beter uitziet en het makkelijker werkt.



Hier een voorbeeld van hoe een class eruitziet. Dit is de guest class met daarin de methods. Hier zie je onder andere de method HomePage().

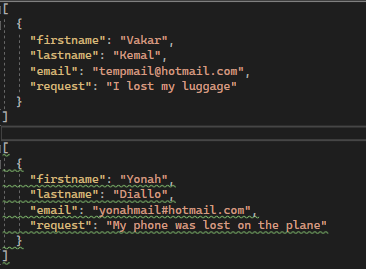


In het begin van ons project maakten we ook gebruik van heel veel if statements. Dit werd ook meteen gecorrigeerd door de product owner en die vertelde ons daarbij dus dat we beter switch cases konden gebruiken. Hier hadden we ons dus meteen in verdiept.

Hier de verbeterde staat waarin we dus switch cases hebben toegepast voor een duidelijker leesbare code.

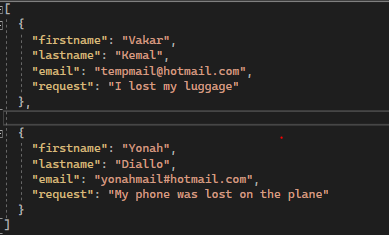
Bij de switch cases krijg je wel een behoorlijk vervelende bug, namelijk dat het programma crasht wanneer je iets anders dan een integer invoert. Dit zorgt ervoor dat het dus geen waterdicht programma is wat natuurlijk wel de bedoeling is.

Daarom hebben we deze functie erin geïmplementeerd. Deze kijkt eerst of het ingevoerde omgezet kan worden van string naar integer. Dit is dus alleen mogelijk wanneer er een getal is ingevoerd. Is dit niet het mogelijk om omgezet te worden crasht het programma niet, maar vraagt het programma opnieuw om een getal. Dit blijft zolang doorgaan tot er een getal wordt ingevoerd.

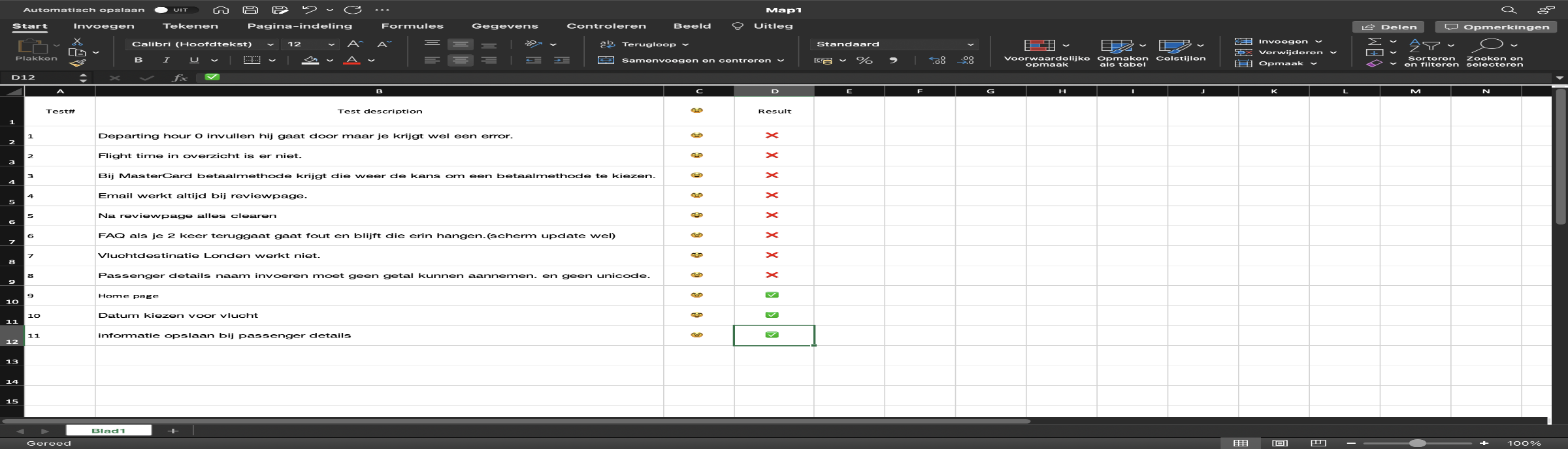




Hierin is te zien dat bij het gebruik van de JSON-files er gegevens verkeerd werden opgeslagen, waardoor we de gegevens niet correct terug konden halen. Toen hebben we dit gefixt door het allemaal in een grote array te zetten. Zoals hieronder te zien is.



Hier zijn de groene lijnen weg wat dus de gefixte versie is.



Ons systeem was getest tijdens de les door Gideon. Hier waren een aantal fouten naar voren gekomen, die we hierboven kunnen zien. Deze hebben we allemaal gefixt in onze code. Het werkt nu allemaal weer prima.