**Integratie Test Specifcatie**

Project: OS3\_HOME\_ALARM

Project team: Groep 5

Teamleden: Lucas Hovestadt

Melvin Kusters

Bas de Jong

Wesley Smits

Opdrachtgever: Fontys Hogescholen

Versie: 0.1

Versiedatum:

Status: Concept

# Documenthistorie

|  |  |  |  |
| --- | --- | --- | --- |
| **Versie** | **Wijzigingen** | **Auteur** | **Datum** |
| 0.1 | Eerste opzet document (Inleiding, strategie en begin van testmatrix / testcases | L.J. Hovestadt | 27/09/2019 |
| **1.0** |  |  |  |

Inhoud

[Documenthistorie 2](#__RefHeading___Toc809_3843518545)

[H1 Inleiding 4](#__RefHeading___Toc811_3843518545)

[H2 Test Strategie 5](#__RefHeading___Toc813_3843518545)

[H3 Test cases en matrix 6](#__RefHeading___Toc815_3843518545)

[H4 Teststrategie 7](#__RefHeading___Toc817_3843518545)

[H5 Logische testcases 9](#__RefHeading___Toc819_3843518545)

[H6 Fysieke testcases 10](#__RefHeading___Toc821_3843518545)

[H7 Testcoverage 11](#__RefHeading___Toc823_3843518545)

[H8 Conclusie *(alleen van toepassing in testrapport)* 12](#__RefHeading___Toc825_3843518545)

# H1 Inleiding

*Het ter ontwikkelen applicatie is een thuis-alarmsysteem. In het kort beschreven zijn er meerdere clients (Arduino uno units + een relay) die zich met de server verbinden. De server beheert communicatie met de clients, maar ook met de eindapplicatie van de gebruiker.*

*Het doel van dit document is om een overzich van onze integratie test strategie te geven en hierbij aan te tonen hoe elk component getest wordt d.m.v. een testmatrix. In het testmatrix en de bijbehorende tabellen zal duidelijk staan beschreven hoe de componenten getest worden en als een geheel moeten functioneren.*

*[1] Zie de SRS voor de requirements van het project en [2] het STS voor hoe deze requirements getest worden. Door het door te nemen van deze documenten zal het duidelijker zijn wat de details van het project zijn.*

*Het concrete doel van integratie testen is dus het zeker stellen dat individuele componenten goed samen werken. Het is niet het doel om de modulen individueel door en door te testen. Zie de gerelateerde documenten hier voor.*

# H2 Test Strategie

*De gekozen test strategie die het beste bij ons project past is de “Bottom-up Integration” test strategie. Zie de twee diagrammen hier onder. [1] Een visuele weergave van hoe een systeem meerdere componenten heeft. [2] De teststrategie en welke testen er eerst worden uitgevoerd.*

main

A

B

C

D

E

F

*Het idee van Bottom-Up Integration is dus dat eerst de hierarchische laagste componenten getest worden, en daar na de componentenlagen hierboven. Wat dit betekend is dat eerst losse componenten getest worden, en daarna geassembleerde en componenten die geinterlinkt zijn.*

*Er zijn een scala aan redenen waarom wij voor deze strategie hebben gekozen. Zoals mogelijk eerder gelezen is in het PMP staat er dat we incrementeel aan het werk gaan in de implementatie fase. Met Bottom-Up Integratie kunnen we [1] parallel met het ontwikkelen van features onze software testen. [2] Onze code en logica grondig doornemen; door incrementeel deel bij deel alles te testen houden we overzicht over het project van nul tot honderd. Hierdoor wordt het ook gemakkelijker om [3] de fouten / bugs op te kunnen sporen. De test methodiek lijkt dan dus ook meer op white-box testing aangezien wij alles door spitten en weten wat er onder de motorkap gebeurt.*

*Verder hebben we te maken met een relatief klein tot middelgroot project. De bottom-up test approach is daar bij goed aangesloten.*

# H3 Test cases en matrix