Bogdanovic J. | Burg van den, R. | Laros Y. | Lynch R. | Veen van, H. Zaalen van, L. |

Fontys | eindhoven

Testverslag

2ND-FActor



**Inhoud**

[**Versiebeheer** 2](#_Toc52184202)

[**1.** **Acceptatietestplan** 3](#_Toc52184203)

[1.1 Milstone 1: Structuur communicatie opzetten 3](#_Toc52184204)

[**2.** **Systeemtestplan** 6](#_Toc52184205)

[2.1 Ststeemplan milestone 1 6](#_Toc52184206)

[2.2 Systeemplan milestone 2 6](#_Toc52184207)

[2.3 Systeemplan milestone 3 6](#_Toc52184208)

[2.4 ToDO 6](#_Toc52184209)

[**3** **Integratietestplan** 7](#_Toc52184210)

[Doelstelling 7](#_Toc52184211)

[Strategie 7](#_Toc52184212)

[3.1 Itegratieplan milestone 1 7](#_Toc52184213)

[3.2 Itegratieplan milestone 2 8](#_Toc52184214)

[3.3 Integratieplan milestone 3 8](#_Toc52184215)

[3.4 ToDO 8](#_Toc52184216)

[**4** **Unittests** 8](#_Toc52184217)

[4.1 Unittest milestone 1 8](#_Toc52184218)

[4.2 Unittest milestone 2 8](#_Toc52184219)

[4.3 Unittest milestone 3 8](#_Toc52184220)

[4.4 ToDO 8](#_Toc52184221)

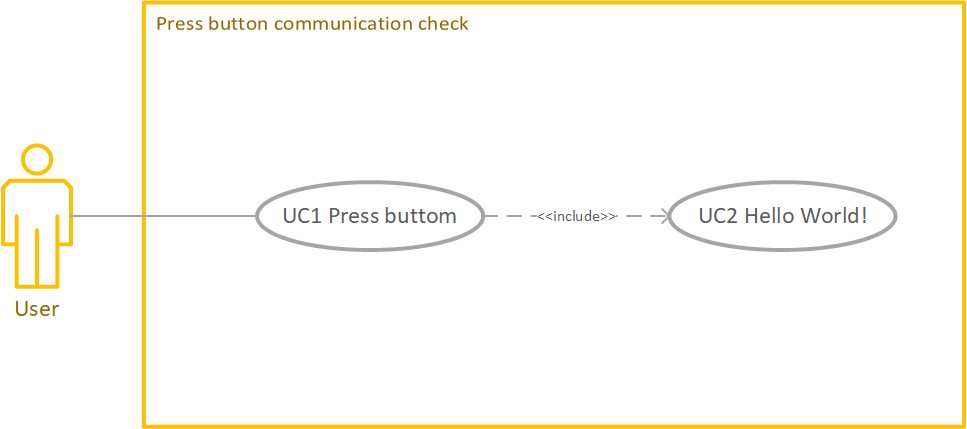
**Versiebeheer**

|  |  |  |
| --- | --- | --- |
| Versienummer | Omschrijving | Toelichting |
| 1.0 | Implementatieverslag | Startdocument |
| 1.1 | Acceptatietest Milestone 1 | Milestone 1 |
|  |  |  |

# **Acceptatietestplan**

* 1. Milstone 1: Structuur communicatie opzetten

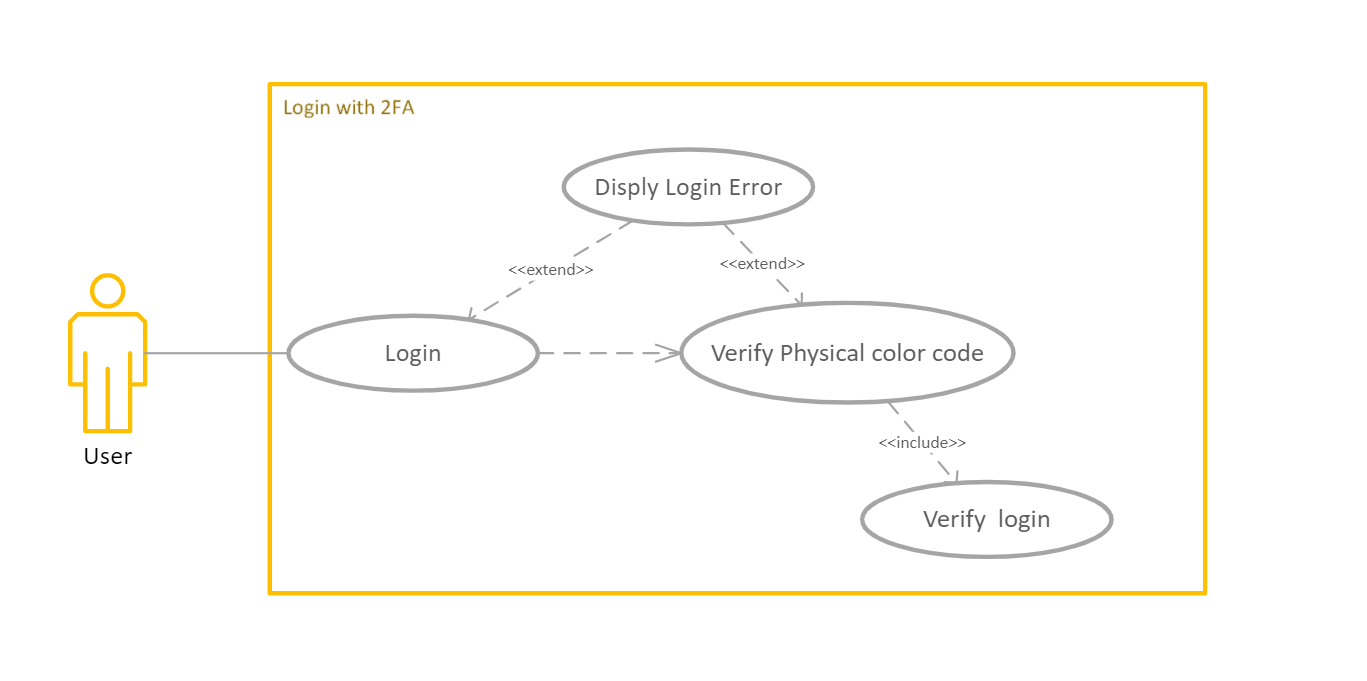
Dit hoofdstuk staat n het kader van het acceptatietestplan. Waarin de applicatie vanuit een userinterface wordt getest. Per milestone is er een usecase diagram gemaakt. Aan de hand van het use-case diagram zijn de verschillende tests opgesteld.



**Hello World printen**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Instructie** | **Verwacht resultaat** | **Werkelijk resultaat** | **Geslaagd?** |
| H-0 | Gebruiker drukt op de “Start” knop. | Een melding met “Hello World” verschijnt. |  |  |
| H-1 | Drukt niet op de knop maar voert enter in. | Foutmelding wordt weergegeven “Druk op start”. |  |  |
| H-2 | Gebruiker voert niets in. | Er gebeurt niets. |  |  |

* 1. Milestone 2: De juiste kleurcommunicatie wordt verzonden
  2. Milestone 3: Beheerpanel opzetten
  3. ToDO
  4. Acceptatietest eindproduct

1. 
2. **Login**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Instructie** | **Verwacht resultaat** | **Werkelijk resultaat** | **Geslaagd?** |
| LO-0 | Gebruiker voert de gebruikersnaam en wachtwoord in. Vervolgens wordt een kleurcode gegenereerd en deze moet in de arduino worden ingevuld. | Een melding met succesvol inloggen verschijnt en de website gaat naar de pagina van de gebruiker. |  |  |
| LO-1 | Gebruiker voert onjuiste wachtwoord in en drukt op inloggen. | Foutmelding wordt weergegeven. |  |  |
| LO-2 | Gebruiker voert onjuiste gebruikersnaam in en drukt op inloggen. | Foutmelding wordt weergegeven. |  |  |
| LO-3 | Gebruiker voert niets in en drukt op inloggen. | Foutmelding wordt weergegeven. |  |  |
| LO-4 | Gebruiker voert onjuiste kleurcode in. | Foutmelding wordt weergegeven. |  |  |
| LO-5 | Gebruiker voert geen kleurcode in. | Een nieuwe code wordt gegenereerd. |  |  |

1. Add new user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Instructie** | **Verwacht resultaat** | **Werkelijk resultaat** | **Geslaagd?** |
| AU-0 | Beheerder selecteert de knop “add user” en voert de gevraagde gegevens in. Hierna wordt op de knop “confirm” gedrukt. | Een melding verschijnt waarbij wordt aangegeven dar de gebruiker succesvol is toegevoegd. |  |  |
| AU-1 | Beheerder selecteert de knop “add user” en voert alleen de noodzakelijke gegevens in, en drukt hierna op de “confirm” knop. | Een melding verschijnt waarbij wordt aangegeven dar de gebruiker succesvol is toegevoegd. |  |  |
| AU-2 | Beheerder selecteert de knop “add user” en voert geen gegevens in en drukt hierna op de “confirm” knop. | Een melding verschijnt dat niet alle noodzakelijke gegevens zijn ingevuld. Tevens worden de noodzakelijke velden met een kleur gemarkeerd. |  |  |

1. Remove user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Code** | **Instructie** | **Verwacht resultaat** | **Werkelijk resultaat** | **Geslaagd?** |
| RU-0 | Beheerder selecteert de een gebruiker en drukt op de “delete user” knop. | Een melding verschijnt waarop wordt aangegeven dat de gebruiker succesvol is verwijderd. |  |  |
| RU-1 | Beheerder selecteert geen user en drukt op de “delete user” knop. | Een melding verschijnt met de vraag om eerst een gebruiker te selecteren. |  |  |

# **Systeemtestplan**

* 1. Ststeemplan milestone 1
  2. Systeemplan milestone 2
  3. Systeemplan milestone 3
  4. ToDo

# **Integratietestplan**

## **Doelstelling**

Het doel van dit integratietestplan is het waarborgen van de functionaliteit vaneen gedeelte van het systeem of een groep systemen in samenhang met elkaar. De verschillende werkende modules worden nu in samenhang en in verbinding met elkaar getest. Hierbij wordt beoordeeld of het geteste gedeelte van het systeem voldoet aan de vooraf gestelde eisen.

## **Strategie**

Voor het bepalen van de integratietest is het bepalen van de strategie de eerste stap. Hierbij kan worden gekozen uit een brede variatie aan strategieën. Voor dit project zijn een aantal populaire strategieën gebruikt als leidraad. De verschillende strategieën zijn als volgt:

* Big-bang

Bij de big bang strategie worden eerst individuele modules getest en vervolgens als een groot geheel.

* Top-down

Bij de top-down strategie worden eerste de hoogste lagen van modules geïntegreerd en getest. Hierna wordt steeds een laag dieper toegevoegd en getest.

* Bottom-up

In tegenstelling tot de top-down methodiek, worden bij de bottom-up strategie eerst de laagste modules geïntegreerd en vervolgens de daaropvolgende hoger gelegen niveaus.

* Sandwich

De sandwich strategie maakt gebruik van de bottom-up en de top-down methodieken en voegt deze samen tot een. Hierbij wordt hert systeem verdeeld in drie lagen waarbij de bovenste laag gebruik maakt van top-down, de onderste laag van bottom-up en de middelste laag afzonderlijk.

* Modified Sandwich

Wederom worden de top-down en bottom-up methodieken gebruikt, waarbij de verschillende modules eerst geïsoleerd worden getest en daarna pas als geïntegreerd systeem.

ToDo

* Risk driven Integration
* Function/thread integration
  1. Itegratieplan milestone 1

# Frontend --- Backend

Uitleg over teststrategie

**-Web page button pressed-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| WBUTTON | Instruction | Expected Result | Actual Result | Succes? |
| WBUTTON \_0.1 | Start de **frontend** en **backend** applicaties. Druk op de **webpagina** op de knop. | De **server** stuurt de **code 200** ter bevestiging van goed ontvangen bericht. Deze code staat in de **console** van de **browser**. |  |  |
| WBUTTON \_0.2 | Start alleen de **frontend** applicatie. Druk op de **webpagina** op de knop. | De **server** stuurt de **code 400** ter bevestiging van niet gemaakte connectie met de backend. Deze code staat in de **console** van de **browser**. |  |  |
| WBUTTON \_0.3 | Start de **frontend** en **backend** applicaties. Druk op de **webpagina** op de knop en schakel tegelijk de **backend** uit. | De **server** stuurt de**code 400** ter bevestiging van niet gemaakte connectie met de backend. Deze code staat in de **console** van de **browser**. |  |  |

**-Server stuurt “Hello World” naar webpagina-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SERVHELLOWEB | Instruction | Expected Result | Actual Result | Succes? |
| SERVHELLOWEB \_0.1 | Start de **frontend** en **backend** applicaties. De **backend** controleert in de **MessageRepository** of er een nieuwe Message aanwezig is. Zorg dat hier een “Hello World” Message in staat. De **backend** pakt de Message met de meest recente timestamp en stuurt deze inhoud naar de **webpagina**. | Op de **webpagina** wordt de boodschap “Hello World” getoond. |  |  |
| SERVHELLOWEB \_0.2 | Start de **frontend** en **backend** applicaties.  De **backend** controleert in de **MessageRepository** of er een nieuwe Message aanwezig is. Zorg dat hier géén Message staat. | De **webpagina** laat géén bericht zien. |  |  |
| SERVHELLOWEB \_0.3 | Start alleen de **backend** applicatie. De server controleert in de **MessageRepository** of er een nieuwe Message aanwezig is. Zorg dat hier een “Hello World” Message in staat. De **backend** pakt de Message met de meest recente timestamp om te versturen. | De **backend** geeft een exception omdat er geen verbinding met de **frontend** gemaakt kan worden. |  |  |
| SERVHELLOWEB \_0.4 |  |  |  |  |

# Backend --- Arduino

**-Arduino maakt verbinding met socket van de backend-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ARDSOCK | Instruction | Expected Result | Actual Result | Succes? |
| ARDSOCK \_0.1 | Start zowel de **Arduino** als de **backend** applicaties. De **Arduino** maakt nu contact met de openstaande socket. | Ter bevestiging van de gemaakte verbinding, gaat de **LED-lamp** op de **Arduino** branden.  In de console van de **Arduino** staat “Connected”. |  |  |
| ARDSOCK \_0.2 | Start alleen de **Arduino** applicatie. De **Arduino** maakt nu geen verbinding met een openstaande socket. | Ter bevestiging van de niet-gemaakte verbinding, blijft de LED-lamp op de **Arduino** uit. |  |  |
| ARDSOCK \_0.3 | Start zowel de **Arduino** als de **backend** applicaties. De **Arduino** maakt nu contact met de openstaande socket.  Schakel hierna de **backend** weer uit. | Ter bevestiging van de gemaakte verbinding, gaat de **LED-lamp** op de **Arduino** branden. In de console van de **Arduino** staat “Connected”.  Zodra de **backend** wordt uitgeschakeld gaat de **LED-lamp** weer uit en staat in de console van de **Arduino** “not connected”. |  |  |

**-De backend stuurt ”Hello” naar Arduino en Arduino stuurt “Hello World” terug naar backend-**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SERVHELLOARDUIN | Instruction | Expected Result | Actual Result | Succes? |
| SERVHELLOARDUIN \_0.1 | Start zowel de **Arduino** als de **backend** applicaties.  Stuur vanuit de **backend** de “Hello” boodschap naar de **Arduino**.  Controleer de **MessageRepository**. | De **Arduino** heeft “Hello World” teruggestuurd. De **backend** heeft deze message opgeslagen in de **MessageRepository** met een timestamp. |  |  |
| SERVHELLOARDUIN \_0.2 | Start alleen de **backend** applicatie op.  Stuur vanuit de **backend** de “Hello” boodschap.  Controleer de **MessageRepository**. | Er is geen nieuwe Message in de **MessageRepository**.  De **backend** applicatie geeft een exception foutmelding in de **console**. |  |  |
| SERVHELLOARDUIN \_0.3 | Start zowel de **Arduino** als de **backend** applicaties.  Stuur vanuit de **backend** de “Hello” boodschap naar de **Arduino** maar schakel tegelijk de **Arduino** uit.  Controleer de **MessageRepository**. | Er is geen nieuwe Message in de **MessageRepository**.  De **backend** applicatie geeft een exception foutmelding in de **backend** **console**. |  |  |
| SERVHELLOARDUIN \_0.4 | Start zowel de **Arduino** als de **backend** applicaties.  Stuur vanuit de **backend** de “Hello” boodschap naar de **Arduino** maar schakel tegelijk de **Arduino** uit.  Controleer de **Arduino console**. | Zodra de **backend** wordt uitgeschakeld gaat de **LED-lamp** weer uit en staat in de console van de **Arduino** “not connected”. |  |  |

* 1. Itegratieplan milestone 2
  2. Integratieplan milestone 3
  3. ToDO

# **Unittests**

* 1. Unittest milestone 1
  2. Unittest milestone 2
  3. Unittest milestone 3
  4. ToDO