

# Technische Acceptatie Testplan

Naam beoordeelde student:

Beoordeeld door:

*De test is tijd gelimiteerd en duurt \$\$\$ minuten. Als de tester niet alle aspecten heeft kunnen controleren houdt dit in dat het voorgaande teveel opmerkingen heeft en dus niet voldoende is, of dat de student handelingen niet snel genoeg kan uitvoeren.*

## 1. IDE

**Test 1.1: Vraag de student een willekeurig bestand van zijn code te openen/tonen in zijn IDE.**

*Observeer daarbij hoe handig de student met zijn werkomgeving omgaat*

Welke IDE wordt gebruikt?

|                              |                                |  |  |  |
|------------------------------|--------------------------------|--|--|--|
| MS Word of "wat is een IDE?" | Notepad, nano of vergelijkbaar | Notepad++, VIM, PSPad of vergelijkbaar | <b>VSCode</b> <sup>1</sup> , Sublime, Brackets, Atom, etc. | Netbeans, Eclipse, PHPStorm/Webstorm, etc. |
|------------------------------|--------------------------------|--|--|--|

Hoe handig is hij en wordt de IDE op de juiste manier gebruikt?

|                                    |   |   |   |
|------------------------------------|---|---|---|
| Lukt niet om het project te openen | Via verkennen dubbelklikken op bestand of vergelijkbaar | Kan met wat geklik een folder of een enkel bestand openen vanuit de IDE | Opent direct de IDE. Van daaruit project openen via <i>recent projects</i> oid. En dan het juiste bestand |
|------------------------------------|---|---|---|

Feedback:

## 2. Codekwaliteit

**Test 2.1: Controleer de folders en bestanden in de projectfolder**

|   |  |  |  |   |
|---|--|--|--|---|
| Rommelige hoop met ook onnodige of juist veel te weinig bestanden | Bestandsnamen die meer iets anders <sup>2</sup> zeggen dan de functie komen voor | url-onvriendelijke <sup>3</sup> bestandsnamen komen voor | Consistent betekenisvolle en url-vriendelijke bestandsnamen en extensies | Voorgaande + gebruikelijke subfolders als img, css, js en lib |
|---|--|--|--|---|

**Test 2.2: Verken in maximaal 3 minuten de code. Benoem alles wat opvalt**

*Geef een score voor de volgende punten. 50% = voldoende; 100% = consistent goed gedaan (zou het zelf niet beter hebben gedaan)*

Aanwezigheid van html, css en javascript?

JA / NEE

*Dit hoeft niet per se in aparte bestanden te vinden zijn in-line css en javascript is ook acceptabel*

Betekenisvolle naamgeving?

*Denk aan CSS-selectoren als classes en id's, javascript functies.*

*Namen moeten iets zeggen over de functie en niet over bijv de vorm (bv een function met de naam 'functie')*

0..100%

Inspringen?

*Elk niveau dieper leidt tot één tab (of 4 spaties) verder inspringen. Onder niveau wordt in ieder geval verstaan een html-child en css declaraties binnen een selector*

0..100%

Paden van afbeeldingen, css-bestanden en andere externe referenties: zijn deze relatief??

*Als ze niet relatief zijn (bv. `http://localhost/img/...` ipv `img/...`) dan is de site niet op een andere locatie te hosten. Als er `file:///...` staat is het helemaal niet goed. Neem de tijd om uit te leggen waarom dat niet goed is*

0..100%

Feedback:

<sup>1</sup> Wordt geadviseerd in deze fase van de opleiding

<sup>2</sup> Bijvoorbeeld 'script.js' of 'styles.css' zijn niet betekenisvol

<sup>3</sup> Dat zijn namen met hoofdletters, spaties en leestekens anders dan '-' en '\_'.

### 3. Testomgeving

| Test 3.1: Laat de student de website demonstreren in de (localhost) testomgeving. Vraag naar welke webserver wordt gebruikt |   |   |   |
|---|---|---|---|
| Via verkenner.<br>URL=file:///...   | Toevallige/standaard localhost webserver of werkt altijd op gehoste omgeving                                | XAMPP (Apache) of vergelijkbaar                                     | Gevirtualiseerde of gecontaineriseerde webserver      |
| Test 3.2: Laat de student zijn gerealiseerde 'CSS-fun' onderdeel uitleggen  |   |   |   |
| Er is geen CSS-fun onderdeel aanwezig   | Er is CSS-fun (werkend of niet), maar alleen copy-paste van internet, en kan niet goed de werking verklaren | CSS-fun onderdeel werkt en kan in grote lijnen de werking verklaren | Uitgebreide CSS-fun. Kan de werking daarvan uitleggen |

Feedback:

### 4. Aanpasbaarheid

| Test 4.1: Laat de student eerst alle bestanden sluiten in de IDE. Bedenk vervolgens een eenvoudige maar in de browser duidelijk herkenbare wijziging in de styling (bv. Achtergrondkleur, of lettertype van navigatiemenu) over alle pagina's. Vraag de student de wijziging uit te voeren en de werking aan te tonen. Vraag hem om steeds 'hardop na te denken' |          |
|--|----------|
| <i>Geef een score voor de volgende punten. 50% = voldoende; 100% = consistent goed gedaan (zou het zelf niet beter hebben gedaan)</i>  |          |
| Student kan handig met de tooling omgaan<br><br><i>Vindt snel de juiste locatie (bestand, regel), vergeet niet op te slaan, etc.</i>   | 0..100%  |
| Werkomgeving is slim ingericht<br><br><i>Er hoeven geen bestanden te worden gekopieerd (projectfolder = documentroot).</i>   | 0..100%  |
| Wijziging hoeft maar op één plaats te worden uitgevoerd<br><br><i>DRY/DIE.</i>   | 0..100%  |
| Test 4.2: Vraag de student de gemaakte wijziging door te voeren naar de productieomgeving ("pushen naar je github repository, en deze als github pages te demonstreren"). Geef wel aan dat de commit herkenbaar moet zijn als testcommit. Vraag hem om steeds 'hardop na te denken'  |          |
| <i>Geef een score voor de volgende punten. 50% = voldoende; 100% = consistent goed gedaan (zou het zelf niet beter hebben gedaan)</i>  |          |
| Student kan handig met de tooling omgaan<br><br><i>Student is bekend met de werkwijze (git add, commit, push). Student handelingen vlot kan uitvoeren (CLI of via IDE)</i>   | 0..100%  |
| Student werkt netjes<br><br><i>O.a. Er wordt een betekenisvolle commit message bedacht of wordt overlegd.</i>  | 0..100%  |
| Gemaakte wijziging is zichtbaar op de hosted omgeving  | JA / NEE |

Feedback: