

Attractiepark Lake Side Mania

Testplan

M1 - Software

Daan Maat, 1213186
3-10-2024

Inhoud

Inleiding.....	2
Teststrategie.....	3
Testcases.....	4
Use case 1.....	4
Use case 2.....	5
Use case 3.....	6
Use case 4.....	7
Use case 5.....	8
Use case 6.....	9
Use case 7.....	10
Componenttest algoritme	11

Inleiding

Dit document is het testplan voor de software ontwikkeld voor het beheer van voorzieningen en voor het genereren van gepersonaliseerde dagprogramma's voor bezoekers. In dit testplan beschrijven we de aanpak, methoden en scope van onze tests. Dit omvat verschillende niveaus van testen - van unit tests en integratietests tot volledige systeemtests - die elk kritische componenten en interacties binnen de software verifiëren.

Teststrategie

Deze teststrategie richt zich op het valideren van de functionaliteit, prestaties en betrouwbaarheid van het software systeem dat ontwikkeld is voor het beheer van voorzieningen binnen een attractiepark en het genereren van dagprogramma's. De strategie omvat verschillende testniveaus, zoals unittests/componenttests, integratietests, en systeemtests, die zijn afgestemd op specifieke use cases.

De verschillende testniveau's zijn als volgt (en volgen het V-model):

Unittests / componenttests

Dit zijn tests die zich richten op individuele componenten of units van de code om te verifiëren dat elk stuk code functioneert zoals verwacht onder gevarieerde condities. Doel: Het identificeren van fouten op het niveau van de kleinste testbare delen van de applicatie, zoals functies of methoden, om zekerheid te bieden over de technische correctheid van de logica. Deze test wordt uitgevoerd op het algoritme, om zeker van te zijn dat de requirements en alle randgevallen goed worden getest en gevalideerd.

Integratietests

Deze tests controleren de connecties en de interacties tussen units / componenten binnen de applicatie. Dit om ervoor te zorgen dat gecombineerde onderdelen van de applicatie correct samenwerken. Specifiek voor hier gaat het om de communicatie en interactie tussen de beheeromgeving en de database en het dagprogramma en de database (met aanvullend hierop de weer API)

Systeemtests

Dit type test controleert het volledige, geïntegreerde software product om te verzekeren dat het voldoet aan de vastgestelde requirements. Doel: valideren dat het volledige systeem functioneert in overeenstemming met de requirements en klaar is voor live gebruik.

Testcases

Use case 1

ID	Beschrijving	Preconditie	Teststappen	Verwachte resultaten
UC1-TC1	Normale functionaliteit voor attracties en winkels/horeca	Database bevat records voor attracties en winkels/horeca .	1. Open de applicatie. 2. Ga naar het overzichtsscherm van de voorzieningen.	Het systeem toont een tabel met alle bestaande attracties. Het systeem toont een andere tabel met alle bestaande winkels en horecagelegenheden .
UC1-TC2	Geen attracties beschikbaar	Database bevat geen records van attracties.	1. Open de applicatie. 2. Ga naar het overzichtsscherm van de voorzieningen.	Het systeem toont de melding: "Geen attracties gevonden."
UC1-TC3	Geen winkels en horecagelegenheid en beschikbaar	Database bevat geen records van winkels en horecagelegenheden.	1. Open de applicatie. 2. Ga naar het overzichtsscherm van de voorzieningen.	Het systeem toont de melding: "Geen winkels en horeca gevonden."

Use case 2

ID	Beschrijving	Preconditie	Teststappen	Verwachte Resultaten
UC2-TC1	Toevoegen van een nieuwe voorziening (attractie)	Database is up-to-date en toegankelijk.	1. Selecteer de optie om een nieuwe voorziening toe te voegen. 2. Vul alle benodigde gegevens in voor een attractie. 3. Selecteer de optie om de nieuwe voorziening op te slaan.	De database bevat de nieuwe attractie. De nieuwe attractie is zichtbaar in de juiste tabel.
UC2-TC2	Toevoegen van een nieuwe voorziening (winkel/horeca)	Database is up-to-date en toegankelijk.	1. Selecteer de optie om een nieuwe voorziening toe te voegen. 2. Vul alle benodigde gegevens in voor een winkel/horecagelegenheid. 3. Selecteer de optie om de nieuwe voorziening op te slaan.	De database bevat de nieuwe winkel/horecagelegenheid. De nieuwe winkel/horeca is zichtbaar in de juiste tabel.
UC2-TC3	Annuleren van het toevoegen van een nieuwe voorziening	Database is up-to-date en toegankelijk.	1. Start het proces om een nieuwe voorziening toe te voegen. 2. Vul enkele gegevens in. 3. Selecteer de optie om de toevoeging te annuleren.	Geen nieuwe voorziening wordt toegevoegd aan de database.
UC2-TC4	Verplichte gegevens ontbreken bij het toevoegen	Database is up-to-date en toegankelijk.	1. Start het proces om een nieuwe voorziening toe te voegen. 2. Laat een of meer verplichte velden leeg. 3. Probeer de nieuwe voorziening op te slaan.	Het systeem geeft een foutmelding over de ontbrekende gegevens. De gebruiker wordt gevraagd de ontbrekende gegevens in te vullen.

Use case 3

ID	Beschrijving	Preconditie	Teststappen	Verwachte Resultaten
UC3-TC1	Voorziening wordt aangepast	Database is operationeel en bevat tenminste 1 voorziening	<ol style="list-style-type: none"> 1. Selecteer de bewerk knop naast de voorziening 2. Verander de waardes die niet meer van toepassing zijn 3. Selecteer de optie om veranderingen op te slaan. 	Het systeem geeft aan dat de verandering is gelukt en de nieuw doorgevoerde aanpassingen zijn zichtbaar in de tabel.
UC3-TC2	Aanpassen van voorziening mislukt	Database is operationeel en bevat tenminste 1 voorziening	<ol style="list-style-type: none"> 1. Selecteer de bewerk knop naast de voorziening 2. Laat 1 van de verplichte velden leeg <p>Druk op de sla wijzigingen op knop</p>	Het systeem geeft een foutmelding met dat niet alle velden zijn ingevuld en vraagt of gebruiker deze velden wil invullen en het dan nogmaals te proberen.
UC3-TC3	Aanpassen van voorziening geannuleerd	Database is operationeel en bevat tenminste 1 voorziening	<ol style="list-style-type: none"> 1. Selecteer de bewerk knop naast de voorziening 2. Druk op de annuleer knop in de pop-up 	Het systeem sluit de pop-up af en er zijn geen wijzigingen toegebracht aan de database.

Use case 4

ID	Beschrijving	Preconditie	Teststappen	Verwachte Resultaten
UC4-TC1	Voorziening wordt verwijderd	Database is operationeel en bevat tenminste 1 voorziening	<ol style="list-style-type: none">1. Selecteer de verwijder knop naast de voorziening2. Selecteer "Ja" op de pop-up die verschijnt wanneer je op de knop hebt geklikt	Het systeem geeft een bevestiging en de voorziening is verwijderd en niet langer zichtbaar in de tabel.
UC4-TC2	Voorziening verwijderen annuleren	Database bevat tenminste 1 voorziening	<ol style="list-style-type: none">1. Selecteer de verwijder knop naast de voorziening2. Er verschijnt een pop-up en selecteer "Nee"	Het systeem sluit de pop-up en de voorziening is niet verwijderd en staat onveranderd in de tabel.
UC4-TC3	Voorziening verwijderen mislukt	Database bevat tenminste 1 voorziening	<ol style="list-style-type: none">1. Selecteer de verwijder knop naast de voorziening2. Er verschijnt een pop-up en selecteer "Ja"	Het systeem geeft een melding dat de operatie niet is geslaagd en dat er iets mis is met de database connectie.

Use case 5

ID	Beschrijving	Preconditie	Teststappen	Verwachte Resultaten
Aanvullen...	Aanvullen...			
Aanvullen...	Aanvullen...			

Use case 6

ID	Beschrijving	Preconditie	Teststappen	Verwachte Resultaten
UC6-TC1	Genereren van een dagprogramma voor bezoekers met standaard en specifieke voorkeuren	Database is up-to-date; JSON-bestanden met standaard en specifieke bezoekersgegevens zijn beschikbaar.	<ol style="list-style-type: none"> 1. Het systeem ontvangt en verwerkt bezoekersgegevens. 2. Het systeem leest voorzieninggegevens in uit de database. 3. Het systeem selecteert toegankelijke voorzieningen. 4. Het systeem selecteert voorzieningen op basis van bezoekervoorkeuren en parkvereisten. 5. Het systeem voegt voorzieningen toe aan het programma totdat de maximale tijd is bereikt. 6. Het systeem slaat het programma op in een JSON-bestand. 	Een gepersonaliseerd dagprogramma wordt gegenereerd en opgeslagen als JSON-bestand, rekening houdend met zowel standaard als specifieke voorkeuren.
UC6-TC2	Afhandeling van ontoegankelijke database	Database is niet toegankelijk; JSON-bestand met bezoekersgegevens is beschikbaar.	<ol style="list-style-type: none"> 1. Het systeem probeert de database te lezen en faalt. 2. Het systeem toont een foutmelding of neemt alternatieve stappen. 	Geen dagprogramma wordt gegenereerd; een foutmelding wordt weergegeven.

Use case 7

ID	Beschrijving	Preconditie	Teststappen	Verwachte Resultaten
UC7-TC1	Weersvoorspelling integratie bij warm weer	Toegang tot real-time weergegevens via een externe dienst.	1. Simuleer een scenario waarbij de temperatuur hoger is dan 20 graden. 2. Controleer of het systeem ijskramen en zomerartikelwinkels aan het dagprogramma toevoegt.	Het dagprogramma bevat suggesties voor ijskramen en winkels met zomerartikelen.
UC7-TC2	Weersvoorspelling integratie bij regenachtig weer	Toegang tot real-time weergegevens via een externe dienst.	1. Simuleer een scenario waarbij de weersvoorspelling regen aangeeft voor de dag. 2. Controleer of het systeem winkels met regenaccessoires aan het dagprogramma toevoegt.	Het dagprogramma bevat suggesties voor winkels met regenaccessoires.
UC7-TC3	Weersvoorspelling integratie bij standaard weer	Toegang tot real-time weergegevens via een externe dienst.	1. Simuleer een scenario waarbij de temperatuur onder 20 graden is en het niet gaat regenen. 2. Controleer het dagprogramma.	Het dagprogramma bevat geen specifieke toevoegingen gerelateerd aan weeromstandigheden.
UC7-TC4	Verbindingsfout met de weersvoorspellingsdienst	Problemen met de verbinding naar de externe weersvoorspellingsdienst.	1. Simuleer een verbindingfout met de weersvoorspellingsdienst. 2. Controleer de reactie van het systeem.	Het systeem toont een waarschuwing over de verbindingfout.

Componenttest algoritme

ID	Testcase
A-TC1	Het systeem moet op basis van de verblijfsduur van de bezoeker en de geschatte wachttijd en doorlooptijd van de attracties, winkels en horeca bepalen hoeveel attracties in het programma worden opgenomen.
A-TC2	Het systeem mag alleen attracties toevoegen aan het programma waar een bezoeker toegang tot mag hebben, gebaseerd op leeftijd, lengte en gewicht
A-TC3	Het systeem moet het programma eerst vullen met attractietypes waar de bezoeker de voorkeur aan heeft gegeven. Attracties mogen maximaal één keer voorkomen, behalve lievelingsattracties die mogen twee keer in het programma voorkomen. Alleen wanneer er ruimte in het programma is, worden andere attracties toegevoegd.
A-TC4	Het systeem moet ervoor zorgen dat de attractietypes gemixt voorkomen in het programma.
A-TC5	Het systeem moet voor iedere bezoeker minimaal één horecagelegenheid toevoegen aan het programma, die aansluit bij de opgegeven voorkeuren voor eten.
A-TC6	Het systeem moet voor bezoekers die langer dan 4 uur in het park verblijven ongeveer iedere 2 uur een horecagelegenheid toevoegen aan het programma.
A-TC7	Het systeem moet voor iedere bezoeker aan het einde van het programma een souvenirwinkel toevoegen.
A-TC8	Het systeem moet via een weer-API kunnen uitlezen wat de temperatuur van de dag is en of het gaat regenen.
A-TC9	Het systeem moet voor iedere bezoeker aan de start van het programma een winkel met zomerartikelen toevoegen, als de temperatuur boven de 20 graden komt.
A-TC10	Het systeem moet minimaal één ijswinkel aan het programma toevoegen, als de temperatuur boven de 20 graden komt.
A-TC11	Het systeem moet voor iedere bezoeker aan de start van het programma een winkel met regenaccessoires toevoegen als het gaat regenen.