

## JSONPlaceholder REST-API – Testaus Robot Frameworkilla ja SoapUI:lla

**Laatija:** Kasper Kallioma

**Päiväys:** 30.10.2025

**Luottamuksellisuus:** julkinen

**Hyväksynyt:** \_\_\_\_\_



## SISÄLTÖ

<b>1</b>	<b>MUUTOSHISTORIA.....</b>	<b>3</b>
<b>2</b>	<b>MÄÄRITELMÄT JA TERMIEN SELITYKSET .....</b>	<b>3</b>
<b>3</b>	<b>JOHDANTO .....</b>	<b>4</b>
3.1	Dokumentin tarkoitus .....	4
3.2	Dokumentin sisältö .....	4
<b>4</b>	<b>TESTAUKSEN KOHDE JA TAVOITTEET .....</b>	<b>5</b>
4.1	Testauskohde .....	5
4.2	Tavoitteet .....	5
<b>5</b>	<b>TESTAUSYMPÄRISTÖ .....</b>	<b>5</b>
<b>6</b>	<b>TESTAUKSEN ORGANISOINTI JA RAPORTOINTI .....</b>	<b>5</b>
6.1	Testausryhmän kokoonpano .....	5
6.2	Raportointi .....	6
<b>7</b>	<b>TESTATTAVAT TOIMINNOT .....</b>	<b>6</b>
<b>8</b>	<b>TOIMINTOJEN TESTITAPAUKSET JA HYVÄKSYMISKRITEERIT .....</b>	<b>6</b>
<b>9</b>	<b>EI –TOIMINNALLISTEN OMINAISUUKSIEN TESTAUS .....</b>	<b>7</b>
9.1	Kuormitustestaus .....	7
9.2	Käytettävyys.....	Virhe. Kirjanmerkkiä ei ole määritetty.
9.3	Siirrettävyys.....	7
9.4	Luotettavuus .....	7
<b>10</b>	<b>OMINAISUUDET, JOITA EI TESTATA .....</b>	<b>8</b>

**1 MUUTOSHISTORIA**

Henkilö	Päiväys	Versio	Kommentti
Kasper Kalliomaa	28.10.2025	0.0.1	Dokumentti luotu
Kasper Kalliomaa	29.10.2025	0.0.2	Dokumenttia muokattu
Kasper Kalliomaa	31.10.2025	0.1.0	Dokumentti jätetty kommentoitavaksi ohjausryhmälle

**2 MÄÄRITELMÄT JA TERMIEN SELITYKSET**

Termi	Kuvaus
API	Application Programming Interface = Rajapinta, jonka kautta eri järjestelmät voivat vaihtaa tietoa
REST	Representational State Transfer = Tapa toteuttaa API, joka käyttää HTTP pyyntöjä tietojen käsittelyyn
JSON	Javascript Object Notation = Helposti luettava tiedonvaihtoformaatti, jota API:n vastaukset käyttävät
Robot Framework	Avoimen lähdekoodin testausautomaatiokehys, jota käytetään mm. API- ja käyttöliittymä testauksessa
SoapUI	Työkalu, jolla voidaan testata REST- ja SOAP-rajapintoja ilman ohjelmointia

### 3 JOHDANTO

#### 3.1 Dokumentin tarkoitus

Tämän testaussuunnitelman tarkoituksena on määrittää, miten JSONPlaceholder REST-rajapinnan toteutus toteutetaan. Suunnitelmassa kuvataan testauksen tavoitteet, laajuus, menetelmät ja testitapaukset. Tavoitteena on testata, että rajapinnan perustoinnot toimivat odotetusti. Lisäksi tavoitteenani on myös testauksen avulla oppia lisää testiautomaatiosta Robot Frameworkin ja SoapUI:n avulla.

#### 3.2 Dokumentin sisältö

Tämä dokumentti sisältää kuvauksen ohjelmistotestausprojektin kohteesta, tavoitteista ja testauksen rajauksesta.

Lisäksi siinä määritellään testauksen toteutustapa, käytettävät työkalut ja testausympäristö.

Dokumentissa esitetään suunnitellut testitapaukset ja niiden hyväksymiskriteerit.

Tavoitteena on varmistaa, että testaus toteutetaan suunnitelmallisesti ja että sen tuloksia voidaan hyödyntää niin oppimisessa, kuin laadun varmistuksessa.

## 4 TESTAUKSEN KOHDE JA TAVOITTEET

### 4.1 Testauskohde

Testauskohteena on JSONPlaceholder REST-rajapinta, joka on julkinen ja suunniteltu ohjelmistokehityksen ja –testauksen harjoitteluun, eikä se edellytä kirjautumista tai API-avainta. JSONPlaceholder simuloi todellisen verkkopalvelun toimintoja, kuten käyttäjien, viestien ja kommenttien käsittelyä.

Testauksessa keskitytään rajapinnan perustoimintoihin, kuten tietojen hakemiseen, luomiseen, päivittämiseen ja poistamiseen. Tavoitteena on varmistaa, että toiminnot palauttavat oikeat vastaukset ja toimivat odotetulla tavalla erilaisissa testitilanteissa.

### 4.2 Tavoitteet

Testauksen tavoitteena on varmistaa, että API:n keskeisten toimintojen toimivan odotetulla tavalla.

Projektin aikana testataan rajapinnan perustoiminnot, kuten GET, POST, PUT, PATCH ja DELETE. Nämä toiminnot testataan API:n eri resursseilla, kuten posts, users ja comments.

Tavoitteena on myös harjoitella ja kehittyä testiautomaation suunnittelussa, toteutuksessa, sekä dokumentoinnissa.

## 5 TESTAUSYMPÄRISTÖ

Testaus suoritetaan omalla Windows-koneella. Kehitysympäristönä käytetään Visual Studio Codea, jossa lisäosana Robot Framework Language Server. Testit ajetaan PowerShellissä. Työkaluina käytetään Pythonia, Robot Frameworkia ja SoapUI:ta

Selain	Käyttöjärjestelmä	Laitteisto	Prosessori
Brave	Windows 11	Windows PC	Intel i5 12400f

## 6 TESTAUKSEN ORGANISOINTI JA RAPORTOINTI

### 6.1 Testausryhmän kokoonpano

Testausprojektin toteutan yksin. Olen vastuussa testauksen suunnittelusta, toteutuksesta, testien suorittamisesta sekä tulosten raportoinnista.

## 6.2 Raportointi

Tulokset testeistä raportoidaan Robot Frameworkin automaattisesti tuottamien report.html ja log.html tiedostojen avulla.

SoapUI:n testitulokset dokumentoidaan ohjelman raporttinäkymästä kuvankaappauksina.

Lopuksi tehdään lyhyt yhteenveto testien kattavuudesta ja tuloksista, joka liitetään projektiraporttiin.

## 7 TESTATTAVAT TOIMINNOT

Use Case	Toiminto	Testaustyökalu
1	Haetaan kaikki postaukset	Robot Framework
2	Haetaan yksittäinen postaus	Robot Framework
3	Luodaan uusi postaus	Robot Framework
4	Päivitetään postaus	Robot Framework
5	Poistetaan postaus	Robot Framework
6	Haetaan kaikki käyttäjät	SoapUI
7	Haetaan yksittäinen käyttäjä	SoapUI
8	Haetaan kommentit tietyistä postauksista	SoapUI
9	Virhe- ja rajatapaukset	Robot Framework

## 8 TOIMINTOJEN TESTITAPAUKSET JA HYVÄKSYMISKRITEERIT

Toiminto	Testitapaus	Hyväksymiskriteerit
Use Case 1	Testataan, että kaikki postaukset saadaan haettua (GET)	Vastaus onnistuu ja saadaan lista postauksista
Use Case 2	Testataan hakea yksittäinen postaus id:n perusteella (GET)	Vastaus onnistuu ja näkyy oikean postauksen tiedot
Use Case 3	Testataan uuden postauksen luonti (POST)	Vastaus onnistuu ja saadaan luodun postauksen tiedot
Use Case 4	Testataan postauksen päivitys (PUT)	Vastaus onnistuu ja kaikki annetut kentät päivittyy

Use Case 4	Testataan päivittää vain osa postauksesta (PATCH)	Vastaus onnistuu ja vain muutetut kentät päivittyvät
Use Case 5	Testataan postauksen poistaminen (DELETE)	Vastaus onnistuu ja tiedot poistuvat
Use Case 6	Testataan hakea käyttäjälista (GET)	Vastaus onnistuu ja listalla näkyy lista käyttäjiä
Use Case 7	Testataan hakea yksittäinen käyttäjä (GET)	Vastaus onnistuu ja haetun käyttäjän tiedot saadaan
Use Case 8	Testataan hakea kommentit postauksen perusteella (GET)	Vastaus onnistuu ja saadaan oikean postauksen kommentit
Use Case 9	Testataan hakea olematonta postausta (GET)	Virheilmoitus tai tyhjä tulos
Use Case 9	Testataan virheellinen pyyntö (POST)	API ei kaadu

## 9 EI –TOIMINNALLISTEN OMINAISUUKSIEN TESTAUS

### 9.1 Kuormitustestaus

Tehdään Load Test SoapUI:lla, jossa samaa pyyntö lähetetään monta kertaa ja varmistetaan että palvelu vastaa pyyntöihin, eikä kaadu.

### 9.2 Siirrettävyys

Testit suunnitellaan niin, että niitä voidaan ajaa helposti eri testausympäristöissä.

### 9.3 Luotettavuus

Testejä suoritetaan eri ajankohtina useita kertoja, jotta voidaan varmistaa API:n tasaisen toimivuus.

## **10 OMINAISUUDET, JOITA EI TESTATA**

Projektissa testaus on rajattu vain API:n perustoimintoihin, joten muun muassa seuraavia ominaisuuksia ei testata.

- Tietoturva
- Suorituskyky
- Ulkoasu ja käyttöliittymä
- Integraatio