

iTEC 2015 - Application Development Challenge

Application Requirements

Why we need this application

You can see UnifiedPost like a large, digital roundabout via which our customers exchange electronic documents, such as e-invoices, fines, salary slips & much more. These documents are fed into our platform via different channels – since we serve a variety of customers with different business models – and we then ensure their successful processing, delivery and legal e-archiving, as well as the integration with electronic ID services, scanning and printing services.

To technically achieve all this, **our applications rely and depend on external web services that often need to be mimicked while testing our applications to make sure we can test any scenario.**

Without it, we can't test every scenario and the tests are slow because each external web service has its own way of configuring an expected response.

What we need

In simple words, we need a **web application that can be trained to respond in a certain way.** That's where you can help us.

Start with a web application using your language of choice. Choose a storage solution. We want to be able to examine what happened in the past and we want to be able to store some settings.

The application user can **generate an unlimited number of pages.**

Each page will **have an automatically generated unique path.**

For each page, **the application user can define a series of parameters that will determine what and how the page is displaying:**

- the HTTP status code to return, required, default 200
- a delay in response, required, default 0
- the response (what the page is displaying) is required and can be empty or text (xml, json, html, etc.)
- a piece of code, optional, that can be dynamically evaluated at runtime to build the response based on the request sent to the page
- authentication credentials required for the page (HTTP Basic, HTTP Headers, OAuth)

When requests are sent to these generated pages, the request and response is stored.

The application user can see the list of generated pages as a list with links to detailed information for each generated page.

The application user **can see the following detailed information for each generated page:**

- request date and time
- request type (GET, POST, PUT, DELETE)
- request body
- response body

Since we're spending time configuring all these pages, we'd like to **protect them with a username and a password**. So **build a user registration and login system that we can use to associate pages to users show only the data that they have created**.

Example use case

As an application user, I open <http://app/create>.

I can see the path was generated automatically to <http://app/use/a3454teh>.

I fill in the following additional data:

- HTTP status code 201
- delay in response to 2 seconds
- response to "Hello World!"

I click a button to save the data that I entered.

I access <http://app/use/a3454teh?code=fun> in a browser.

I check <http://app/requests> and I can see a list with one element, <http://app/use/a3454teh> and a detail link.

When I click on the detail link, I see the following:

- the date and time when I accessed <http://app/use/a3454teh?code=fun> in the browser
- GET
- code=fun
- Hello World!

Appearance and Design

The application should be very simple and easy to use. It should maintain the basic look and feel of our www.unifiedpost.com website, reduced to a very limited colour scheme of white, grey and orange (colour codes as per corporate website).

Useful links & additional information:

Client-server model

<http://simple.wikipedia.org/wiki/Client-server>

Representational state transfer

http://en.wikipedia.org/wiki/Representational_state_transfer

HTTP status codes

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

Server application example

<https://github.com/cmer/postbin.rb>

<http://sourceforge.net/p/soaprest-mocker/wiki/Home/>

<https://www.mockable.io/>

<http://requestb.in/>

Evaluation Criteria & Scoring

Below is an overview of how the points are distributed. Please note that the number of points indicated for each category below **constitute the maximum score** - you can be awarded 1 out of 5 or 5 out of 5 in each case, depending on how well your application answers the requirements.

Functional features

Total **100 points**, distributed as follows:

- the page path can be generated automatically and is unique – **4 points**
- authentication credentials for the page can be set for a page – **4 points**
- the scanned response can be set for a page – **3 points**
- HTTP status code can be set for a page – **3 points**
- delay in seconds can be set for a page – **3 points**
- dynamically evaluated code can be set for a page – **3 points**
- the page responds as configured without dynamically evaluated code – **10 points**
- the page responds as configured with dynamically evaluated code – **20 points**
- the request to a page is stored – **10 points**
- the response for a request is stored – **10 points**
- there is a page to see a list of generated pages – **5 points**
- in the list of generated pages, each page links to a detailed page – **5 points**
- on the detailed page, you can see the request date, type and body and the response – **10 points**
- the user can register – **3 points**
- the user can login – **3 points**
- the user can see only the pages generated by him – **4 points**

Added value

Total **50 points**, distributed as follows:

Maintainability & clean code - **20 points**

UI, UX – **15 points**

Originality, how the app is presented to the judges – **15 points**

Procedure for Mentoring

- An email address has been created: itec2014.appdevelopment@gmail.com
- Mentors will monitor this email address in the intervals mentioned in the schedule below;
- Since the teams will be working in parallel, there is a good chance the same questions will be addressed multiple times. To address that, we have created a shared Google doc where all questions asked by participants & their answers will be copy / pasted.
- Please keep checking this file, as questions & answers are added to it in real time:
https://docs.google.com/document/d/13ikbHpoD2nml7cbvB3rdtNi3vBqzwMfrlc_uh6gfVqM/edit?usp=sharing

Availability of Mentors

- Mentors will be available to answer your questions in the interval **14.00 – 19.00 on Friday, 3 April 2015** and **10.00 – 16.00 on Saturday, 4 April 2015**.

Rules for answering participants' questions

- **We will answer** questions about the functional side of the application;
- **We will not answer** questions asking for concrete technical solutions (e.g. how should I create this login?)