Camel App Use Cases

### Introduction

Camel app is a wrapper around certain components provided by camel framework. This app will be directly called from the browser as a rest call.

### Terminology

User: The individual who is using the camel app or the user who is developing an app which uses this camel app.

Developer: The individual who is developing the camel app.

End User: The individual who is using the developed app which in-turn uses camel app.

### Scope

The scope of this document is to consider/evaluate various use cases which Camel app has to handle.

### Methodology

The user who would like to use Camel app has to first understand how to use camel components.

The documentation of camel components are available at

<http://camel.apache.org/components.html>

Out of all the buffet of components that are in place the user would like to use “pdf” component of camel, the following is the documentation link of the same.

http://camel.apache.org/pdf.html

For every camel component, there are certain operation’s which user can perform and various options which user can pass while performing the operation. In the case of “pdf” component the operations are

Create

Append

extractText

The URI that has to be constructed to use this component is

pdf:operation[?options]

So in the case of “pdf” component, the URI that will be formed to create a pdf document is

pdf:create?marginTop=20&marginBottom=20

For every camel component, we will also have headers, which can be configured and passed while calling the component.

To use any kind of Camel component we first create the URI’s and headers of the components, The URI construction for an app under a context remains the same, so this URI can be templated and stored in AppProperties, the values that are required for the template will be passed as request params.

Incase of this “pdf” document the URI that will be constructed in App Properties will be

Key: pdf

Value: pdf:$operation?marginTop=$request.body.params.top&marginBottom=$request.body.params.bottom

On the browser, we can construct request as follows.

Method: POST

URL: <https://myapp.appup.cloud/camel>/execute/

{

app: pdf,

operation: createPdf,

top: 20,

bottom: 20

}

In this request app and operation, params are mandatory

### Use cases :

**Camel App config.json :**

{

**"workflows"**: [

{

**"id"**: 22542,

**"name"**: **"generic"**,

**"trigger"**: **"rest"**,

**"expression"**: **"generic"**,

**"method"**: **"POST"**,

**"steps"**: [

{

**"uri"**:**"{{{request.body.uri}}}"**,

**"value\_type"**: **"hb"**,

**"body"**: **"{{{request.body.message}}}"**,

**"output\_variable"**: **"output"**,

**"type"**:**"camel-custom-step"**,

**"headers"**:**"{{{request.body.headers}}}"**,

**"id"**: **"1"**,

**"next"**: {

**"success"**: **"3"**,

**"failure"**: **"hangup"**

}

},

{

**"variable\_name"**: **"output"**,

**"http\_response\_code"**: **"200"**,

**"response\_content\_type"**: **"text/html"**,

**"value\_type"**: **"hb"**,

**"type"**: **"send"**,

**"id"**: **"3"**,

**"next"**: {

**"success"**: **"hangup"**,

**"failure"**: **"hangup"**

}

}

]

}

],

**"filters"**: [],

**"server"**: {

**"port"**: **"8081"**,

**"host"**: **"127.0.0.1"**

},

**"plugins"**: [

]

}

To hit Camel app we have to pass body parameters and api call to hit camel app like below config.json

{

**"workflows"**: [

{

**"name"**: **"rest-simple-step"**,

**"id"**: 200,

**"trigger"**: **"rest"**,

**"expression"**: **"restsimple"**,

**"method"**: **"GET"**,

**"steps"**: [

{

**"type"**: **"start"**,

**"id"**: 1,

**"next"**: {

**"start"**: **"start"**

}

},

{

**"url"**: **"http://127.0.0.1:8081/generic"**,

**"method"**: **"post"**,

**"headers"**: **""**,

**"output\_variable"**: **"output"**,

**"value\_type"**: **"hb"**,

**"body\_parameters"**: **"{uri:\"aws-s3://appup-bucket?region=US\_EAST\_1&accessKey=AKIAIVJMBGBK4PRJ3SKA&secretKey=RAW(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)\",message:\"Hello world\" , headers:\"CamelAwsS3Key=test/ghhjj.txt ,CamelAwsS3BucketName=appup-bucket\"}"**,

**"type"**: **"appup-notification-step"**,

**"id"**: **"2"**,

**"next"**: {

**"success"**: **"PBXevPcWDyO8q"**,

**"failure"**: **"hangup"**

}

},

{

**"variable\_name"**: **"output"**,

**"http\_response\_code"**: **"200"**,

**"response\_content\_type"**: **"application/json"**,

**"value\_type"**: **"hb"**,

**"type"**: **"send"**,

**"id"**: **"PBXevPcWDyO8q"**,

**"next"**: {

**"success"**: **"hangup"**,

**"failure"**: **"hangup"**

}

}

]

}

],

**"filters"**: [],

**"server"**: {

**"port"**: **"8080"**,

**"host"**: **"127.0.0.1"**

},

**"plugins"**: []

}

### 