## Message Resources

Message Resources is a tile for both authenticated and unauthenticated user.

### Functional Overview

### Visual Design

#### Unauthenticated / Anonymous Experience



#### Authenticated Experience



### Assumptions

The request parameters/data required to invoke the API’s will be available in the session/context.

The Page Manager App will be providing the user context using which the authenticated or unauthenticated user’s data will be resolved by the CRF template/Tile Service.

### Tile APIs

|  |  |
| --- | --- |
| Sign In | |
| Method Name | getMessageResources |
| URI | /aries-common/ messageResources |
| Local Endpoint URL | http://{ipAddress}:{port}/aries-common/ messageResources |
| Variations | 0.1 |
| Default Variation | 0.1 |
| HTTP Method | POST |
| Request Format | JSON |
| Query Parameters | N/A |
| Response Format | JSON |
| Sample Request | {  "context": {  "localeKey": "en\_US",  "siteId": "",  "siteName": "marriott.com",  "pageContent": [{  "contentName": "couples\_page",  "type": "page"  }]  },  "variation": "0.1",  "sessionToken": "06A5A231-AE8A-530D-B76C-DC6E4794B204"  } |
| Sample Response | {  "status": "success",  "message": "Common Component successfully resolved.",  "messageCode": "ACA-SR-01",  "sessionToken": "06A5A231-AE8A-530D-B76C-DC6E4794B204",  "component": {  "name": "messageResource",  "template": "messageResource",  "bundles": [],  "data": {  "name": "msgResources",  "properties": {  "prevMonthText": "Previous month",  "weekdayShortList": "[\"Sun\",\"Mon\",\"Tue\",\"Wed\",\"Thu\",\"Fri\",\"Sat\"]",  "nightText": "night",  "loadingText": "Loading...",  "clearText": "Clear",  "nightsText": "NIGHTS",  "weekdayFullList": "[\"Sunday\",\"Monday\",\"Tuesday\",\"Wednesday\",\"Thursday\",\"Friday\",\"Saturday\"]",  "nextMonthText": "Next month",  "monthShortList": "[\"Jan\",\"Feb\",\"Mar\",\"Apr\",\"May\",\"Jun\",\"Jul\",\"Aug\",\"Sep\",\"Oct\",\"Nov\",\"Dec\"]",  "NewWindowAltText": "Open in a different browser tab",  "closeText": "is-hidden-text>this window</span>",  "doneText": "Done",  "monthFullList": "[\"January\",\"February\",\"March\",\"April\",\"May\",\"June\",\"July\",\"August\",\"September\",\"October\",\"November\",\"December\"]",  "printText": "Print"  }  }  }  } |
| Error Codes | TBD |

### Sequence Diagram

[Refer Section 4.1.4.1](#_Maken_–_Reservation) depicts the application flow when a request is made by MAKEN to get the details for the Message Resources tile.

#### Narrative

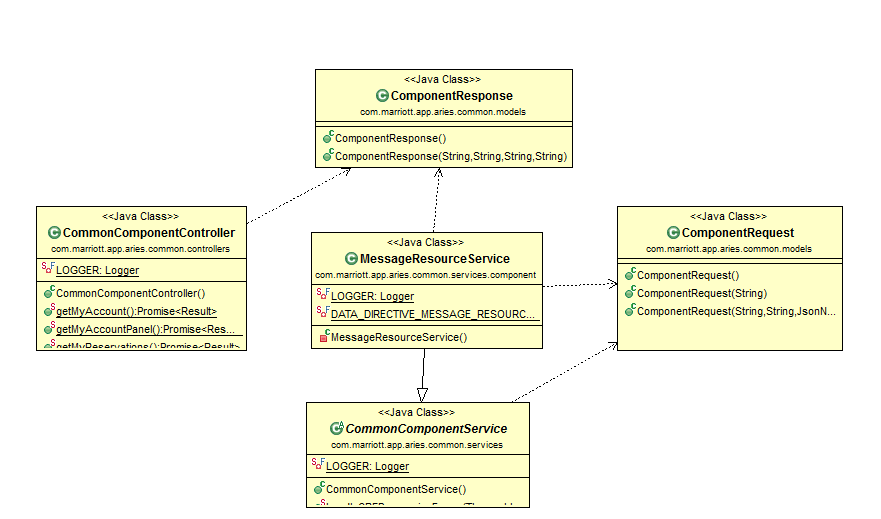
1. MAKEN parses the HTML page (from the presentation app) content to identify the tile markers
2. For the common tile marked as server side rendering, MAKEN makes a REST service call using the location URI available as part of the marker.
3. The tile details REST service call goes to NGINX proxy server
4. NGINX routes the request for the common tile to the Common Play App
5. The request is handled by the CommonComponentController.
6. The CommonComponentController calls MessageResourceService.
7. The MessageResourceService processes the context data (passed along by MAKEN from the marker) and request information (like locale) to determine the version and variant of the tile to be loaded.
8. The MessageResourceService calls ComponentManager which invokes the CouchbaseClient to retrieve the tile metadata / JSON template details from Couchbase.
9. The ComponentManager processes the JSON template for applying the variation and the corresponding options and overrides available for that variation.
10. The ComponentManager invokes BindingManager to resolve the directives present in the CRF JSON template.
11. The BindingManager invokes the ENTERPRISE API’s via the label and controller list directives to resolve the values.
12. The BindingManager aggregates the template details, tile messages, content and data to create the tile details JSON response.
13. The MessageResourceService wraps tile details JSON to ComponentResponse.
14. The common app serves the REST service call back to MAKEN via NGINX with the tile details (JSON).

### Tile Configuration

| **Description** | **Configuration** |
| --- | --- |
| Couchbase Tile Configurations | couchbase.header.menu.componentId="MessageResource"  couchbase.header.menu.version="v1" |
| Content API Configurations | # Common headers for all calls to Content API  content-api.headers.Authorization="Basic cmFtOjJjR3FtdGhIWWg2eEZ6NXJSc3lBcmxvUWZ6SXJDd05zeWpHSUFOZg=="  content-api.headers.X-Device-ID="MARRIOTTDOTCOM"  content-api.headers.Content-Type="application/json"  # Override default timeout for Content API calls  content-api.timeout-ms=20000  # Configuration for account-information Label API  content-api.account-information.url=${content-api.labels.base-url}"/account-information"  content-api.account-information.response-element-name="contents"  # Configuration for paymentdetails-component Label API  content-api.paymentdetails-component.url=${content-api.labels.base-url}"/paymentdetails-component"  content-api.paymentdetails-component.response-element-name="contents"  # Configuration for profiledetails-component Label API  content-api.profiledetails-component.url=${content-api.labels.base-url}"/profiledetails-component"  content-api.profiledetails-component.response-element-name="contents"  # Configuration for Reservation component Label API  content-api.reservation-component.url=${content-api.labels.base-url}"/reservation-component"  content-api.reservation-component.response-element-name="contents"  # Configuration for search-component Label API  content-api.search-component.url=${content-api.labels.base-url}"/search-component"  content-api.search-component.response-element-name="contents" |

### Class Diagram

The diagram below depicts the classes that are created for this tile.



#### Class Names

| **S.No** | **Java Classes** | **Class Description** |
| --- | --- | --- |
| 1 | CommonComponentController | Controller receives the request from NGINX and calls the service class responsible for handling tile request. |
| 2 | CommonComponentService | Base service class which has basic implementation for common methods to be used by all the services. |
| 3 | MessageResourceService | Service class populate the data in the model and sends the model to Platform for CRF resolution. |
| 4 | ComponentResponse | Common Tile Response JSON will be mapped to this Object |
| 5 | ComponentRequest | Common Tile Request JSON will be mapped to this Object |

### Rendering Logic-Legacy

### Business Rules

### API Integration

#### Content API

| **Label Key** | **Bundle** | **Arguments** | **Value** |
| --- | --- | --- | --- |
| close.modal.text | account-information |  | is-hidden-text>this window</span> |
| loading.text | profiledetails-component |  | Loading... |
| print.this.reservation.text | reservation-component |  | Print |
| night.text | paymentdetails-component |  | night |
| nights.text | account-information |  | NIGHTS |
| clear.text | search-component |  | Clear |
| done.text | account-information |  | Done |
| open.new.browser.tab | account-information |  | Open in a different browser tab |

#### Functional API - Tile Attributes & Source

Not Applicable

### Session Data

#### RAM Session Data

| **Attribute Name** | **Description** |
| --- | --- |
| Context | Temporarily kept in session until actual source is confirmed. |

#### Legacy Session Data

Not Applicable

### Legacy Application Changes

#### Legacy Code Changes

### CRF Template

|  |
| --- |
| **MessageResource\_v1.json** |
| *{*  *"componentName": "messageResource",*  *"componentDescription": "Show all the hidden messages of a page.",*  *"bundles": [],*  *"defaultVariation": "0.1",*  *"variations": {*  *"0.1": {*  *"templateName": "messageResource",*  *"template": [*  *"messageResource.handlebars"*  *],*  *"options": {},*  *"overrides": {}*  *}*  *},*  *"responseTemplate": {*  *"properties": {*  *"closeText": "@label{account-information|close.modal.text}",*  *"loadingText": "@label{profiledetails-component|loading.text}",*  *"printText": "@label{reservation-component|print.this.reservation.text}",*  *"nightText": "@label{paymentdetails-component|night.text}",*  *"nightsText": "@label{account-information|nights.text}",*  *"clearText": "@label{search-component|clear.text}",*  *"doneText": "@label{account-information|done.text}",*  *"NewWindowAltText": "@label{account-information|open.new.browser.tab}",*  *"prevMonthText": "@config{previous.month}",*  *"nextMonthText": "@config{next.month}",*  *"monthFullList": "@data{messageResources.monthFullList}",*  *"monthShortList": "@data{messageResources.monthShortList}",*  *"weekdayFullList": "@data{messageResources.weekdayFullList}",*  *"weekdayShortList": "@data{messageResources.weekdayShortList}"*  *},*  *"name": "msgResources"*  *}*  *}* |

### Exception/Error Handling

#### Platform Exception/Error Handling

Below are general error/exception scenario(s).

| **#** | **HTTP Status Code** | **Error Code** | **Error Message** |
| --- | --- | --- | --- |
| 1 | 500 | SERVER-ERROR-PA-008 | Entry not found in cache server |
| 2 | 500 | SERVER-ERROR-PA-010 | JSON Parsing Exception while processing the request |
| 3 | 500 | BAD-REQUEST-PA-001 | Session token missing in request |

#### Tile Exception/Error Handling

### Issues/Constraints

The API contracts have not been finalized yet