**FINAPP APIs design review – recommendations:**

API Products : FINAPP APIs

API Version: v1

Environment: QA (new Apigee migrated APIs)

Review date: June 17, 2022

Reviewed By: Platform Enablement – API design enablement team.

**API design review scope**:

The API is reviewed under API design “standard” category and the review didn’t include individual specific data element naming and data structure standards.

There are 2 APIs included in the design review scope.

1. FinappGuxApigee
2. FinappsAPApigee

Note: The 2 APIs are not following proper REST standards and will fail basic Linter checks. We should run the Linter report and share those basic violation reports also to the team.

**API Implementation design review**:

Overall observation:

There are several basic issues with the APIs design but I still reviewed and provided high-level feedback despite the fact that these APIs may fail linter validations.

The design review feedback is included below.

**API documentation**

API Catalog ref: API catalog don’t have new swagger files published yet and I reviewed the swagger files received from the PDO team and included them here for reference.



**Recommendations (not in any order or priority):**

**Common review feedback items across all APIS:**

1. Both the APIs do not follow REST resource and endpoint definitions and using verbs like “createDealJacket” in endpoint definitions are against REST design standards. Strongly recommending team to review the API style guide (reference below) and make necessary changes.
2. Please do not use gateway names like “apigee” and environment names like “qa”, “prod” in API META data definitions. This is against standards. Also, please publish the QA APIs in QA API catalog (currently I see the entry in PROD API catalog for both QA and PROD).
3. Create and manage Deal Jacket is Ford credit integration with RouteOne and not sure why MSS team developed a proxy interface for the same rather than the service provider FMCC to manage them for all channel needs as they own the credit app implementation with RouteOne.
4. There are already duplicate Deal Jacket and credit app status APIs in API catalog and these new APIs seems to be developed to migrate them towards Apigee. Please remove the old Deal jacket APIs from API gateways and API catalogs to avoid any confusion.
5. I also see an opportunity to consolidate both the APIs as they all related to create Deal jacket, credit app and the status and do not see need for 2 separate APIs. This change is better than 4 APIs from legacy world but still there is an opportunity to reduce it to 1 API. Team can continue to maintain multiple micro services if they want to but API exposed at API gateway level can be consolidated at minimum.
6. Base paths: Please reconsider the base path names and make sure they are human readable and follow Ford API style guide recommendation of splitting each work with “-“.
7. The error codes shall follow Ford style guide standards and shall not include attribute level specific validation errors.
8. Lack of API documentation for both the APIs and there are no clear descriptions of what each API endpoint is doing. Not easy to understand for people outside of PDO team. Also, please upload the latest swagger files into API catalog (even to QA) with updated descriptions.

**API specific Review feedback – FinappGuxApigee:**

1. Please follow generic overall recommendations provided above.
2. Endpoints: please do not use words with verb+noun words for resources and endpoints (crateDealJacket, initiateCreditApp are like RPC style design). Please make sure the API endpoint is clearly defined as a plural noun resource applicable and include endpoint attached to that specific resource.
3. Please also remove county code as part of endpoint definition. Country of US can be sent as HTTP header using ISO standard 3 digit or 2-digit country code and country code shall not be part of endpoint definition as it not associated to any resources but provide only context.
4. I am providing few endpoint examples below but team can use better naming conventions for resources and endpoints which better suits their business needs (examples only to provide context)
   1. POST “/finance/credit/applications/deal-jackets” 🡪 create deal jacket
   2. POST “/finance/credit/applications/reference-url” 🡪 provide US reference URL. country = “US” to be included as part of HTTP header.
   3. Since the request data is not confidential, please think about using GET method to return URL by passing attributes via Query parameters as well.
5. Don’t understand all endpoints returning both 200 and 201 HTTP response codes? I can understand one of them (ex: 201 may be best suited for create Deal jacket) but having 2 success error codes seems to be copy/paste issues? Also, 404 error doesn’t seems to be applicable for this API and please fix them. Also, please follow Ford API style guide to include Ford API error code definition standards (reference below).
6. Also, for create Deal jacket endpoint, don’t we get some reference like Deal Jacket reference ID or so in the response? Why the swagger has empty response object? Not sure if I am missing something?

**API specific Review feedback – FinappsAPApigee:**

1. Please follow a) generic overall recommendations above b) design review recommendations in those lines of FinappGuxApigee (would like to avoid duplicating the recommendations).
2. Already provided feedback above for not using verbs+ nouns in endpoints (updateOffer, sendContract status like RPC style design ). I am providing few endpoint examples below but team can use better naming conventions for resources and endpoints which better suits their business needs (examples only to provide context)
   * GET “/finance/credit/applications/deal-jackets/{dealJacketID}/status?dealerID=xxxx” 🡪 returns deal jacket status. Conversation ID can be passed as HTTP header and that’s the right place for it (RouteOne also uses conversation ID in HTTP headers). DealerId can be passed as URL query parameter.
     + But if Deal jacket ID is unique, why do we still need to pass Dealer Id in query parameter?
   * If you want to use POST the way it is today, please consider the following
     + POST “/finance/credit/applications/deal-jackets/status”
   * Also, don’t see the response object definition for contract status response and it is empty.
     + Please use enumeration codes for contract status and define it in the response body structure for better API design standards.
3. Update offer endpoint:
   * Please don’t pass county code as part of request object and please use ISO standard 2 or 3-digit country code and include it in HTTP header and that’s the right place for it for proper API design implementations.
   * I still don’t understand what this endpoint is returning. URL? But this can be simply GET endpoint with an example below
     + GET “/finance/credit/applications/deal-jackets/{dealJacketID}/offer-url?orderId=xxxx” . We can work out the final endpoint design with team inputs and this is a quick example for reference.
4. Error codes: same issue as above API feedback.

We can help the team for proper final design of the API when team is ready to make the changes.

**References:**

**API Security Review**:

References document:

<https://www.eesewiki.ford.com/display/FEAPIS/API+Security+with+42Crunch>

**API Style guide Review**:

References document:

<https://www.eesewiki.ford.com/display/FEAPIS/Style+Guides>