Pilot’s API Design Review Checklist

**System and Process API's**

Must-have:

API has version, title and description

API has a well populated documentation section (Home + Legal)

API supports http and https protocols

API uses traits and resourceTypes

API Resource URL follow /collection/entity pattern

API is protected by at least basic auth or clientid and clientsecret

API uses fragments to describe the conveyed resource objects

All operations have a request (body, http headers, uriParameters, queryParam) example All responses have a response (body, http headers, uriParameters, queryParam) example Support application/json

Error response based of the general template "rest-error"

Examples are externalised in the API Specification in a clean folder structure. Should-have:

Gzip output support for payloads > 100KB

Pagination support on collections where the number of results (unfiltered) > 100 Description for each operation (GET/PUT/POST/PATCH/DELETE/HEAD/..) Examples are objects, not JSON or XML

Could-have:

All object elements are explicitly typed

**Experience API's**

Must-have:

API has version, title and description

API has a well populated documentation section (Home + Legal)

API supports http and https protocols

API is protected by at least basic auth or clientid and clientsecret

All operations have a request (body, http headers, uriParameters, queryParam) example All responses have a response (body, http headers, uriParameters, queryParam) example Examples are externalized in the API Specification in a clean folder structure.

Should-have:

API uses fragments to describe the conveyed resource objects

API uses traits and resourceTypes

API Resource URL follow /collection/entity pattern

Gzip output support for payloads > 100KB

Pagination support on collections where the number of results (unfiltered) > 100 Description for each operation (GET/PUT/POST/PATCH/DELETE/HEAD/..)

Examples are objects, not JSON or XML

Could-have:

Support application/json

All object elements are explicitly typed

Error response based of the general template "rest-error"