Today we are starting the second iteration of testing. Lets get started.

0 – Phase

1 – Test Data

| **Name** | **Value** |
| --- | --- |
| {PostOffice\_ID} | 131000 |
| Tariff zones (full list) | 1, 2, 3, 4, 5 |
| Time zone (full range) | 1 – 12 |
| {Tariff\_Version} | 11.1.5 |
| {FrankMachine\_ID} | PM100200 |
| {Contractor\_ID} | 0008d2cd-549e-11ea-80e8-00155d027901 |
| {Contract\_ID} | 00000177 |

2 – Integration Requirements

| **Requirement type** | **Implementation** |
| --- | --- |
| Authorizing requests SDU → ASU FM | Every request that is sent must contain a header **X-API-Key**, where apiKey - unique identificatory.  **X-API-Key**value: postmartUat. |
| Returning errors when accessing the SDU | In case of any error on the side, the SDU must respond with the following data structure     |  | | --- | | {    "code": 1000, // unique code    "message": "Readable message"  } |     “code” - is a unique numeric error code  “message” -  is a readable error message    **HTTP error codes:**   * *400 Bad Request* - business errors (wrong entity ID, wrong parameters) * *404 Not Found* - when requesting to get an entity by a specific identifier * *500 Internal Server Error* – other errors |

3 – Restrictions

1. Order of the phases: 1 → 2
2. Order of the steps: 1 → 2 (phase 1) and 1 → 11 (phase 2)
3. Inside the step, there is also the order of passing the cases starting from 1.
4. If the step is unsuccessful, the next step cannot be started.
5. On step 2-Step (Taxes data sending) there is an example of a request with a limited number of postal products. In fact, the number of postal products> 200 (248) and can be supplemented and changed according to the requirements of the Russian Post.