DEFECT REPORT

| **Defect ID: ID\_001**  **Description:** As per the POST Product schema definition, when the attribute **product\_price** is sent in the request body, 400 error code is received.  **Raise By:** | **API Name:** POST/product  **Env**: Local  **Build Version**:v1 |
| --- | --- |
| **Steps to reproduce:**   1. Execute POST product request in swagger.   {  "name": "primrose",    "product\_price": 9.25  }   1. Error message is received- 'price' is a required property | |
| **Expected Result:**  Response code 204 should be displayed but error is displayed. Schema definition should read ‘price’ and not product\_price | |
| Evidence | |

| **Defect ID: ID\_002**  **Description:** POST product endpoint returns 204 successful code when already existing product\_code is also provided in the request but code is auto generated  **Raise By:** | **API Name:** GET/product  **Env**: Local  **Build Version**:v1 |
| --- | --- |
| **Steps to reproduce:**   1. Execute POST product request in swagger.   {  "name": "primrose",  "product\_code": 14,  "product\_price": 9.25}   1. Response code 204 is returned | |
| **Expected Result:**  Response should show a bad request error but successful 204 is returned. This is because the product code provided in the request is not validated. | |
| Evidence | |

| **Defect ID: ID\_003**  **Description:** GET product endpoint is returning ‘id’ in the response body instead of returning ‘product\_code’  **Raise By:** | **API Name:** GET/product  **Env**: Local  **Build Version**:v1 |
| --- | --- |
| **Steps to reproduce:**   1. Execute GET product request in swagger by providing the product code 2. Response body is returned   {  "id": 5,  "name": "Jasmine",  "price": "9.25"  } | |
| **Expected Result:**  Response body should return product\_code instead of returning the ‘id’ or the schema has to be updated | |
| Evidence | |

| **Defect ID: ID\_004**  **Description:**As per schema definition price is a number but in the response it is returned as string  **Raise By:** | **API Name:** GET/product  **Env**: Local  **Build Version**:v1 |
| --- | --- |
| **Steps to reproduce:**   1. Execute GET product request in swagger by providing the product code 2. Validate the Response body is returned   {  "id": 30,  "name": "extra large",  "price": "9.25"  } | |
| **Expected Result:**  Price should be a number as per the schema definition | |
| Evidence | |