National Informatics Centre

E Invoice

Error handling for effective and efficient use of APIs

As all are aware, NIC is running the E Way Bill and E Invoice Systems smoothly since the launch. Though NIC has the sufficient infrastructure to handle large number of transactions per day, as well as large number of transactions simultaneously during peak hours, close look at the transaction logs show that there have been large number of API calls that are unsuccessful due to certain errors. This large number of unproductive API calls will affect all the stake holders by consuming the compute resources and the network bandwidth and affecting the performance. The carefully designed system resources can be effectively utilized by reducing / nullifying the errors. Most of the operations through APIs are happening through the GSPs and hence this document is being shared with all the GSPs.

We have been monitoring and reaching out to the tax payers by way of automated emails and in some cases, we are contacting the tax payers and appraising them of the errors generated by them and how to avoid them. With this, the erroneous API calls from those tax payers have completely stopped. This indicates that the issue could be handled by writing proper code or fixing the bugs in the consuming application. We have also published the cause and resolution of the errors in the developer’s portal.

To mitigate this issue and to provide smooth and efficient services to the genuine API calls, we are working out a strategy to identify the top error generators, intimate the concerned tax payer about the issue, allow them to correct the same within certain period. If the issue is not resolved within the allowed period, the system would automatically block the access to the E Invoice System for that tax payer.

Basic requirement of API integration is logging the request and responses and monitoring the same. If the API call has succeeded, a flag has to be set so that the same payload is not sent again, if the call has failed, the response should be analyzed, required corrections have to be applied and only then the payload has to be sent to the server. Purpose of providing the error details is to facilitate necessary action to be taken before accessing the API again, to get the job completed. It is observed that the same erroneous request payload is used many number of times without applying the required correction. This will never solve the problem but consume the resources unnecessarily

Some of the top errors, the pattern of their occurrences are being highlighted here for the benefit of the GSPs so that they can educate their respective clients to effectively and efficiently use the API system.

1. **Duplicate IRN:** One of the common errors is Duplicate IRN, with error code 2150. As it is not possible to generate multiple IRNs for the same combination of supplier GSTIN, doc. Type, doc. No. and fin. Year, once the IRN is generated, subsequent request to generate IRN with same request will throw an error saying duplicate IRN. For the benefit of the user, in case they have failed to receive the response when the IRN is generated for the first time, subsequent attempt to generate IRN with same payload would return the details of the IRN already generated. Using this information, one can get the complete details of the E Invoice.

Some of the scenarios have been shown in the following table

Number of requests with same payloads (Same requests, different dates; Same date, different users; Same user, different requests)						
S. No	**GSP**	**Taxpayer**	**Date**	**Request**	**Count**	**Remarks**
1	GSP1	GSTIN1	14-12-2020	Request 1	43*	Same tax payer using same request payloads on multiple days
2			15-12-2020	Request 1	90	
3			16-12-2020	Request 1	91	
4			17-12-2020	Request 1	92	
5			18-12-2020	Request 1	92	
6			19-12-2020	Request 1	91	
7	GSP2	GSTIN1	19-12-2020	Request 1	253	Different GSTINs under same GSP making multiple requests with same as well as different payloads.
8			19-12-2020	Request 2	161	
9		GSTIN2	15-12-2020	Request 1	11	
10			16-12-2020	Request 1	21	
11			16-12-2020	Request 2	9	
12			19-12-2020	Request 1	14	
13		GSTIN3	15-12-2020	Request 1	334	
14			15-12-2020	Request 2	333	
15	GSP3	GSTIN2	16-12-2020	Request 1	27	Same GSTIN sending different payloads multiple times on the same day
16			16-12-2020	Request 2	47	
17			16-12-2020	Request 3	45	
18			16-12-2020	Request 4	45	
19			16-12-2020	Request 5	42	
20			16-12-2020	Request 6	53	
21			16-12-2020	Request 7	54	

*Annexure A

Resolution:

- Once the IRN is generated, the response should be saved and a flag should be updated against the invoice to indicate that the IRN is generated against this document and should not be used again to call the API.
- Even if the user is not able to receive the response after getting the IRN generated, the second attempt itself will provide the IRN details along with the error response. The response includes the IRN number, time of generation of IRN, act. No. The IRN number from the same may be used to get the further details of the E Invoice using the Get IRN details API
- So, there should not be more than 2 requests with same request payload to generate the IRN

2. **Duplicate requests:** It is observed that there are simultaneous requests (received at the server exactly at the same instance of time). This will create an application error. Since each request is a new request at that point of time, duplicate check passes and system tries to save all IRNs and then it fails to save other request data except one. Similar issue is applicable for authentication requests where separate tokens will be generated resulting in 'Invalid Token' error.

Simultaneous hits (Different GSP, Tax payer, Date and Payloads)					
S. No	**GSP1**	**GSTIN1**	**Date**	**Request Count**	**Count**
1	GSP1	GSTIN1	19-12-2020 17:35:10	Request 1	2
2		GSTIN 2	21-12-2020 00:52:29	Request1	2
3	GSP 2	GSTIN 1	14-12-2020 15:26:53	Request 1	2
4		GSTIN 2	16-12-2020 13:57:17	Request 1	2
5		GSTIN 3	16-12-2020 13:57:26	Request 1	2
6		GSTIN 4	16-12-2020 13:57:32	Request 1	2
7		GSTIN 5	16-12-2020 13:59:13	Request 1	2

Resolution:

- Take care of not to fire same request payloads simultaneously.
- Though the server responds within a second, it is always advisable to use same request payload after few seconds.
- If multiple client applications or multiple client systems or threads are being used, care should be taken to avoid picking same payload by more than one application or system or thread.

3. **Inactive GSTIN or GSTIN not available**: Multiple requests with same payload having same inactive GSTINs / non existing GSTINs. The API continues to give the same error response unless the GSTIN is corrected or the respective GSTIN data is synced between GSTN Common portal and E Invoice System.

No. of transactions by a GSTIN on sample dates		
Date	GSTIN	Count
23-12-2020	GSTIN1	23222
27-12-2020	GSTIN1	20379
30-12-2020	GSTIN1	8152

Resolution:

The user has to check the correctness of the GSTIN used and check whether the GSTIN is active on the day of the document date. If yes, still the problem continues, the GSTIN has to be synced in E Invoice system from the GSTN common portal using the following alternatives

- In case there is a response with error code 3028 or 3029, the correct ness of the GSTIN should be checked on the E Invoice portal -> Search -> Tax payer and if required use the option “Update from GST Common Portal”
- Use “Get GSTIN Details” API to check the availability and status of the GSTIN and if required, use the “Sync GSTIN details from Common Portal” API to sync the data between GSTN and E Invoice system for the given GSTIN

4. **PIN Code does not belong to given State Code**: The PIN code given does not belong to the state provided. The correct PIN codes and State code combination should be provided wherever applicable

No. of transactions by a GSTIN on sample dates		
Date	GSTIN	Count
23-12-2020	GSTIN1	14408

28-12-2020	GSTIN1	15470
30-12-2020	GSTIN1	14468

Ex.:

1. "Pincode -846008 of Buyer does not belong to his/her State - 6 "
   -- State code should have been 10
2. "Pincode -670101 of Buyer does not belong to his/her State - 29"
   -- State code should have been 32
3. "Pincode -635109 of Buyer does not belong to his/her State - 29"
   -- State code should have been 33

Resolution:

- State wise PIN code list, as per E Invoice database is available for download in MS Excel file at E Invoice portal -> Search -> Master Codes.
- The same may be downloaded incorporated in the user application
- User has to send correct combination of PIN code and State Codes

5. **Invalid HSN Code**: Many requests are received where the HSN code is wrong or does not exist.

   Ex.:

   1. "HSN code(s)-39200000 is invalid” **Should have been 3920**
   2. "HSN code(s)-73170000 is invalid" **Should have been 7317**
   3. "HSN code(s)-94054099,94054000,85371099,39269000 is invalid” Wrong format

   Resolution:

   - The list of HSN codes used in the E Invoice system are available at E Invoice portal -> Search -> Master Codes.
   - The HSN code should be sent as it is shown in the master list.
   - There should not be prefixing zeros, if it is shown as 4/6 digit code, four/two more zeros should not be suffixed to make it 8 digit which does not exist in the master etc.

Annexure A

Time at which same request was received from same GSTIN on same date for generation of IRN resulting in error response with code 2150

S. No	GSP	GSTIN	Request	Time of access
1	GSP 1	GSTIN 1	Request 1	14-12-2020 12:44:08.307
2				14-12-2020 12:59:19.263
3				14-12-2020 13:14:38.603
4				14-12-2020 13:29:50.110
5				14-12-2020 13:46:46.600
6				14-12-2020 14:03:08.637
7				14-12-2020 14:18:28.963
8				14-12-2020 14:33:49.747
9				14-12-2020 14:49:08.977
10				14-12-2020 15:04:36.470
11				14-12-2020 15:20:07.033
12				14-12-2020 15:36:15.673
13				14-12-2020 15:52:11.570
14				14-12-2020 16:07:23.977
15				14-12-2020 16:22:34.980
16				14-12-2020 16:37:45.987
17				14-12-2020 16:54:12.867
18				14-12-2020 17:24:46.593
19				14-12-2020 17:40:58.200
20				14-12-2020 17:56:29.670
21				14-12-2020 18:11:37.043
22				14-12-2020 18:26:49.803
23				14-12-2020 18:42:09.033
24				14-12-2020 18:57:15.960
25				14-12-2020 19:12:24.083
26				14-12-2020 19:27:29.790
27				14-12-2020 19:42:35.783
28				14-12-2020 19:57:40.243
29				14-12-2020 20:12:44.350
30				14-12-2020 20:27:48.170
31				14-12-2020 20:42:49.927
32				14-12-2020 20:57:51.660
33				14-12-2020 21:12:56.467
34				14-12-2020 21:28:00.093
35				14-12-2020 21:43:02.163
36				14-12-2020 21:58:07.053
37				14-12-2020 22:13:12.370
38				14-12-2020 22:28:16.850
39				14-12-2020 22:43:18.890
40				14-12-2020 22:58:22.227

41				14-12-2020 23:28:46.793
42				14-12-2020 23:44:22.193
43				14-12-2020 23:59:24.417