



Application Programming Interface Specifications

Integrated e-filing and CPC 2.0 Project

API Name	getPrefill
API Description	API used to get Prefill data

Version History

Version	Date	Description
1.0	29-10-2021	Initial Draft
1.1	17-11-2021	Exception scenarios added

Table of Contents

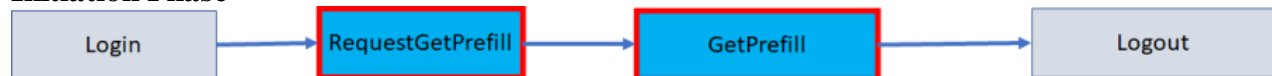
1. Overview.....	4
2. About API.....	4
3. Target Audience and Pre-requisites.....	4
4. RequestPrefillOTP API Details.....	5
4.1 API Usage Scenario	5
4.2 API Request process.....	5
4.3 API Protocol.....	5
4.3.1 Request Header:	6
4.3.2 Request Body:.....	6
4.3.3 Details of data attribute:.....	6
4.4 Response Parameters.....	7
4.5 requestPrefillOTP API - Sample Request format.....	8
4.6 requestPrefillOTP API - Sample Response format.....	8
5. getPrefill API Details.....	8
5.1 API Usage Scenario	8
5.2 API Request process.....	9
5.3 API Protocol.....	9
5.3.1 Request Header:	9
5.3.2 Request Body:.....	9
5.3.3 Details of data attribute:.....	10
5.4 Response Parameters.....	11
5.5 getPrefill API - Sample Request format	12
5.6 getPrefill API - Sample Response format.....	12
6. API Exception Details.....	13

1.Overview

As noted in the ERI Specification Overview document the interaction during the initiation phase involves fetching the prefill information for the taxpayer. This document describes the API specification for the same.

Note: The requests for get prefill data will be tracked and measured for this API response will be applicable to block and legal action.

Initiation Phase



2.About API

Requester	Type-2 or Type-3 ERI
Provider	ItrWeb
Description	These APIs shall be used to fetch prefill data for the taxpayer. There will be two calls to get Prefill data. One is to request for the prefill data where OTP will be sent to taxpayer and second call is to get prefill data after providing OTP value by the taxpayer.
Mode of Integration (Real time / Batch)	Real Time
Processing Details	Fetches the prefill data as per the schema published in the income tax portal ERI can fetch prefill data only for current assessment year 2021
Pre-Processing Details	User should have established a session using the Login API specifications.
Service Name	requestPrefillOTP getPrefill
API URL	TBD

3.Target Audience and Pre-requisites

This is technical document and is target to ERIs working in their application and interested to integrate their application with IEC 2.0 platform.

The pre-requisites to call this API is that ERI is already registered with IEC 2.0 platform. They have valid credentials to call the API.

4.RequestPrefillOTP API Details

This service is used by ERI to request for getPrefill of taxpayer. This API will submit the request from ERI and send OTP to taxpayer to get consent. The ERI can get prefill data for taxpayer only when the consent is given by the taxpayer by entering OTP. Request for prefill can be performed only when taxpayer is already added as its client.

4.1 API Usage Scenario

This API is called taxpayer is already registered to eFiling system and he/she is already added as client of ERI. Where ERI wants to submit a request to get prefill data from the eFiling system for their client.

When ERI submits request to get prefill for their client, there are two high level activities are performed:

1. Client (taxpayer) will get OTP on her/his mobile and email address. OTP will be sent from eFiling system to primary mobile number and primary email address of the taxpayer or Aadhaar OTP will be sent depending upon the option provided in API request.
2. ERI application requests the taxpayer to provide the OTP in their application. This OTP value should be used to get the taxpayer's consent and pass in getPrefill API request.

Above two steps are two disconnected API calls and OTP value sent from using requestPrefillOTP API which is step 1 above is then used as part of the getPrefill API to get the taxpayer's consent and get the prefill data.

4.2 API Request process

ERI application must have already establish the login session and auth token with eFiling system using the login API. Application will initiate requestPrefillOTP request as below:

1. ERI application will capture the taxpayer's details as explained as part of the request attribute details.
2. Application will initiate request to requestPrefillOTP API to submit the request

4.3 API Protocol

requestPrefillOTP API is exposed as REST API over the HTTPS. The input data should be sent as JSON document using Content-Type "application/json".

The request will consist of request header and request body:

4.3.1 Request Header:

Header is mandatory and will consists of following values:

Mandatory Request Header Parameters:

Header Name	Header Value
Content-type	application/json
clientId	clientId value which is provided to ERI as part of the registration
clientSecret	clientSecret value which is provided to ERI as part of the registration
authToken	Auth token from the Login Flow
accessMode	"API"

4.3.2 Request Body:

Request body will consist of below attributes:

1. **data:** data attribute will be Base64 encoded string of API request json. Details of request json attributes are explained in request data element details.
2. **sign:**
 - The API request data attribute should be digitally signed for the message integrity and non-repudiation purposes.
 - Digital signing should always be performed by the ERI from value of data attribute which was generated from request json.
 - The signature should be generated using a valid X.509 certificate
 - signature value should be generated from data field using ERI's DSC private key.
 - ERI should share their DSC public key with ITD to validate the signature.

3. **eriUserId:** It is mandatory and valid value is user ID of the ERI

4.3.3 Details of data attribute:

Below are the request parameters, which is request json used to create data attribute as explained above data attribute of the request body:

Name of the Parameter	Data type	Max length	Is Mandatory	Description
-----------------------	-----------	------------	--------------	-------------

serviceName	String	60	Yes	It is mandatory and valid value is "EriGetPrefill"
Pan	String	10	Yes	Valid PAN of the taxpayer whose prefill information is sought
assessmentYear	String	4	Yes	Assessment year for which the prefill information is sought
otpSourceFlag	String	1	Yes	OTP Options user has selected for getting consent from the taxpayer: Valid values are: "E" for eFiling OTP "A" for Aadhaar OTP

4.4 Response Parameters

Name of the Parameter	Data type	Max length	Is Mandatory	Description
messages	List<String>		Yes	List of information messages. There can be one or more messages. The messages array will be null if there are no messages. ["messages": [] This is an array which has 4 sub parameters – code, type, desc, fieldName
code	String	7	yes	Error/message code depending on validation response
type	String	10	yes	It describes type of message
desc	String	50	yes	It describes Error/message if validation is passed/failed
fieldName	String	50	no	It describes the field name, when not applicable null will be returned
errors	List<Error>		Yes	List of errors. The value will be null if there is no error. "errors": []
successFlag	Boolean		Yes	Indicates whether requestPrefillOTP call is successful. Possible value is true or false. "successFlag": true

transactionId	String	20	Yes	Unique number generated
HttpStatus	String	20	Yes	http codes corresponding to response. Possible value is "SUBMITTED" or "REJECTED" "HttpStatus": "SUBMITTED"

4.5 requestPrefillOTP API - Sample Request format

```
{
  "data": "",
  "sign": "",
  "eriUserId": ""
}
```

data tag will be Base64Encoded string from following request json

```
{
  "serviceName": "",
  "pan": "",
  "assessmentYear": "",
  "otpSourceFlag": ""
}
```

4.6 requestPrefillOTP API - Sample Response format

```
{
  "messages": [],
  "errors": [],
  "successFlag": true,
  "transactionId": "1324t56",
  "HttpStatus": "SUBMITTED"
}
```

5.getPrefill API Details

This service is used by ERI to provide taxpayer's consent and get prefill for the taxpayer.

5.1 API Usage Scenario

This API is called taxpayer is already registered to eFiling system and he/she is already added as client of ERI. Where ERI wants to provide taxpayer's consent with OTP value and get prefill

data from the eFiling system.

When ERI provides OTP value from taxpayer and get prefill for them, there are two high level activities are performed:

1. Taxpayer will get the OTP based on the request was made to requestPrefillOTP API earlier.
2. Taxpayer will provide the OTP values as their consent.
3. API will provide the prefill data based on the consent given by the taxpayer.

5.2 API Request process

ERI application must have already establish the login session and auth token with eFiling system using the login API. Application will initiate getPrefill request as below:

1. ERI application will capture the taxpayer's OTP values as their consent.
2. Application will initiate request to GetPrefill API to get prefill data for the taxpayer.

5.3 API Protocol

GetPrefill API is exposed as REST API over the HTTPS. The input data should be sent as JSON document using Content-Type "application/json".

The request will consist of request header and request body:

5.3.1 Request Header:

Header is mandatory and will consists of following values:

Mandatory Request Header Parameters:

Header Name	Header Value
Content-type	application/json
clientId	clientId value which is provided to ERI as part of the registration
clientSecret	clientSecret value which is provided to ERI as part of the registration
authToken	Auth token from the Login Flow
accessMode	"API"

5.3.2 Request Body:

Request body will consist of below attributes:

data: data attribute will be Base64 encoded string of API request json. Details of request json attributes are explained in request data element details.

sign:

- The API request data attribute should be digitally signed for the message integrity and non-repudiation purposes.
- Digital signing should always be performed by the ERI from value of data attribute which was generated from request json.
- The signature should be generated using a valid X.509 certificate
- signature value should be generated from data field using ERI's DSC private key.
- ERI should share their DSC public key with ITD to validate the signature.

eriUserId: It is mandatory and valid value is user ID of the ERI

5.3.3 Details of data attribute:

Below are the request parameters, which is request json used to create data attribute as explained above data attribute of the request body:

Name of the Parameter	Data type	Max length	Is Mandatory	Description
serviceName	String	60	Yes	It is mandatory and valid value is "EriGetPrefill"
Pan	String	10	yes	Valid PAN of the taxpayer whose prefill information is sought
assessmentYear	String	4	yes	Assessment year for which the prefill information is sought
otpSourceFlag	String	1	Yes	OTP Options user has selected for getting consent from the taxpayer: Valid values are: "E" for eFiling OTP "A" for Aadhaar OTP
mobileOtp	String	6	Yes	OTP value for Aadhaar OTP or mobile OTP value for eFiling OTP
emailOtp	String	6	No	It is mandatory only in case otpSourceFlag i.e., "E". Email OTP value

TransactionId	String		Yes	Transaction ID which was received from previous service call to requestPrefillOtp
---------------	--------	--	-----	---

5.4 Response Parameters

Name of the Parameter	Data type	Max length	Is Mandatory	Description
messages	List<String>		Yes	List of information messages. There can be one or more messages. The messages array will be null if there are no messages. "messages": [] This is an array which has 4 sub parameters – code, type, desc, fieldname
code	String	7	yes	Error/message code depending on validation response
Type	String	10	yes	It describes type of message
desc	String	50	yes	It describes Error/message if validation is passed/failed
fieldName	String	50	no	It describes the field name, when not applicable null will be returned
errors	List<Error>		Yes	List of errors. The value will be null if there is no error. "errors": []
successFlag	Boolean		Yes	Indicates whether GetPrefill call is successful. Possible value is true or false. "successFlag": true
transactionId	String	20	Yes	Unique number generated

prefill	json		Yes	Prefill data json. It will be encrypted which can be decrypted using the key shared with ERIs. Kindly refer to the JSON schema for the prefill information published on the income tax portal
---------	------	--	-----	--

5.5 getPrefill API - Sample Request format

```
{
  "data": "",
  "sign": "",
  "eriUserId": ""
}
```

data tag will be Base64Encoded string from following request json

```
{
  "serviceName": "",
  "pan": "",
  "assessmentYear": "",
  "otpSourceFlag": "",
  "mobileOtp": "",
  "emailOtp": "",
  "transactionId": ""
}
```

5.6 getPrefill API - Sample Response format

```
{
  "messages": [],
  "errors": [],
  "successFlag": true,
  "transactionId": "1324t56",
  "Prefill": "kqAQF7c0wTJ3z94pAv8wQQ7eZxXSPLYkqQP8tPq2QywUz5mLvWK4k78FHW8wL1aZ6BbbKMH3ygCJrVDLFeZX6yOvbSj6Jdfh8/ifo82hKg9M2yYeab5tg0vHzqDveXVsCqe7u7oLX9DZA9a4rTrMM/zM7Y98dJjhEpzvnoalWIDEBDrfuAXDuN5JwLSYiDkKZhiekkYY+EKO6GHtYPrca1sL9B310Aciq0ixiEPKvt5l7czQTm4luNLAD6TIWhuQVfe1wp5fydj9S6f1CmqIzI319Jh42SGFSpjRQ+KwqsnppTdY2/q6O8CvYt0MebwytWHadTL4RIIEqENFdR1r+ScgVkmGyX/M5jMDuY6NhZNRlgRwogT5cbe/d2C15vwNm3WNeqmnhRgZwNnPYdoUjoN4cG6LCPfPNZ5PF829JyQSoA55Xt1XrlKVz00gDwzzCFGz7Rsf3csCAKI10efwt"
```

1lpX + NdYKHjBYhPaR0dg2iabIN8tm843HuzG7833I22IMqdahxH4B3r0PI6qn3iiQodMLpNXBc6miJvMLF4pyrSn4KmYA
 Q8BmM0cqqiCtq9lj2PO3xe8HP4eMbdw45cCxRCfRd53B2AAzS9 + xYmanlkl/NgRkFSwRBreVgWlv + TYkwx2brbg7N1qO
 ulCCD0UGrBvKkVYxUXJqwhcm2pZx1sMIACKYQW5KdzR4ZHE4fl2w + OlobgiT/68xbFZPM4CDpxuhB7oEIUvphiCuXkQkL
 7B5Csaipzck9kJdq8FnQrFYGE1A9qzagAQevSpZulPlsEMryj8ZK5ke7R8MfNvh3KzUT4/Fzic4zzJ82ELnQj0HOMByw42sO
 myUFgLHki + D81pREyLiO7EowlsphvFMz6YhVaU6Ro3c7LtyVG7Xvs4XDek/Hwg3MmGd2DUTI51TRA7MXcmKEj/RIEN9aj
 8FDQkTazya7YoScBN2qk4CJYFa73wsLpgOFeMhXbDiMeEEHQZrjZIVUJeybDDWe/Lt + 8DwtpeBQcN/I6WLnlzTh2Opcsil
 3V0pz67BK + JcCuPxtgCxFw13dGIXsDn6itBhp/wQN9yPjHW5A8 + dW2f3gaRbjTkVY5CUg3xcfoyScb/7IBcTIfuil5E37Y3UPr
 VSjZwAO3jsDKbxTDLmOCYBtkq1/a0O6VbtTgKam4Zb/rvs6hwxnzop0NdAkcSIWpISWGYsrHW + qdu5JStK86c945f0Uk
 kBR9Z8Gdt5krwO0ITlaDLNz3NPsKbEU6uVw3PTiWsDiUy8OgiJ KSTsnPjeIXjN1/XJtGoOvOneIn1opwLWrs2PDQWO + cur
 ehoylJES6mNI43YAOHHFdBS9hqC + fgMYPRdxIESdlbzWTF3cApu1J4Bs84GSAUyRs1NAuzFVnfmFiqwsj8qrUd9Dj5izMp
 qv3h3NGBYfCpFOUN43MFpp7CRsAL21o9wqpfyBVfaJR8FnmwZfRkqwuzmZwTQVnhNjK59Nv/uTqZqLamZqgU0z2Z1K
 dXMG0vJzVn5Cj0jBB0DfZ/Gc8qMb/hE3L9IkY8hB1R6426Cf/IZLAV3wMORQ2MoD/pW0QtO8fAd7tMftMVSvlfvY/Dljksu
 IEkmJDgdy1lfcMwicqyJ8JKImCWZa2RZBBEdFE6EzSXNkQQxHbJdDE9hzMXPvshgaN2anaTdDmOcb4CSiA4FJl3urz0PY0
 3zHIQW85IU8vdLGGtG/9RJAd5Y5g + oDoy1/m0Z/rz1xJzvtRmKAHclyqDfWqPS/oaEQnz7p60nhus4HdAsJNG4z4NmHm
 EjjL6OilakEzRk1JSE7Y + 9OvqsxYYhZBV7jzOcf2o4MgOBSzd7q89D2NNxyEfvOg3FJw8aGjUG7pmW4/NYvD0H9ubbjOj
 GES0GvJimtERE8qldy/JwHEgnRPqpTtxwO1gcuBXO + EjSaUuado8bMKLC5DHZFltjymwRBWgfQzj4b/PyP7FmPKgtyXMw
 R2BgINvDIIEUvYGo4wWYOSEpyNgcOfU50xcNbVYnMXbhwpS/pqS + gJn0Zs9JzA3K9JNDywlUqzWJIQYD7wc6SfNIs8Rl9
 CetsaYk7h9nm6sn0oMewWoSFuoqiWu + TZV6XZOMRI/oJwdFiE3P0z73vcuHBXpS1FXK1begYckbGxwypFUuSkavfRiYE
 pBtGsSQw1SQFjH/7LalfPeP5vFSYASWtjmcarij5xc9bejjvu2T2Qm3bcZF1OmyFqHtj70n4yxj1OTs3/3RXI0SQu8YHYwiRa40t
 m92fe41747 + ETDKVqnVBokfokG6KvLUjADchOyCn7s26BVu0VifWegdILrHH4efDVbKm5g8jMU/lpXGXNaMjMbi5jgYrq
 UATT + EpNQS7NnEbV10CieRuVg1 + dF9p9kDFEzUx2U62/LzXmfseM3N2imHsFEjf5m6QLp6zn + WC9mCCoc/lW4MU + Jij
 upwzXMXrh7Cz/5V1MBD7O2pi749SyPONtI7TkOxH1fpiZgnVkmCnvnniChX84jqtWxfa7A6iBYh5X187liw15T2Yl8wWlfoW
 D1sLGMWY6666ewaDt + MdpJBlwq2oRiEbclz0d4FDZQuXFn + ancU6rhSVhWoe47oElanJeX4Hcts/26xEOEzSf67QNzHZw
 HWoeX5cV3FIdaobYuW72ShmCV5I0ua0Rt8DtaFWBlsgCAFbUaPRch/do3D2HA6o0Bn/kXx/JHEPpAbKWlfr + e2nEE7tG
 m13tI0Q79pPjUcHOETxbNZmFasJHSuDieIwIvJwfc + NByh0oXeySj/V23YPfVAtZof6Zj6qtsiK4h2BBgOnJorH + zGRnl4and
 rp6gj12fShEL + eHeMcm/tM6Y4Mx0DBc = ”

}

6.API Exception Details

Scenario	Error Code	Error Message	Message Type
When PAN is not in a valid format	EF00011	Please enter a valid PAN Number.	ERROR
When Pan Entered does not exist.	EF00047	The PAN does not exist.	ERROR
When OTP is sent successfully based on given request	EF40010	OTP has been sent successfully.	REMARK

When OTP generation fails	EF40014	OTP Generation failed.	ERROR
When any attributes are missing in request JSON	EF40000	JSON data invalid.	ERROR
When user does not enter OTP number.	EF00014	Please Enter OTP Number.	ERROR
When the OTP is incorrect.	EF00016	The OTP has expired.	ERROR
When email OTP entered is incorrect.	EF00072	The Email OTP you have provided is incorrect. Please retry	ERROR
When mobile OTP entered is incorrect.	EF00073	The Mobile OTP you have provided is incorrect. Please retry	ERROR
When OTP entered is incorrect.	EF00096	The OTP you have provided is incorrect. Please retry	ERROR
When user does not enter email OTP number.	EF500033	Please enter email OTP.	ERROR
When user does not enter mobile OTP number.	EF500032	Please enter mobile OTP.	ERROR
When user does not enter Aadhaar OTP number.	EF500034	Please enter Aadhaar OTP.	ERROR
When user does not enter OTP source flag.	EF500031	Please enter OTP source flag.	ERROR
When AY is enter wrong or previous year.	EF500052	Please select correct AY. Prefill can be done through this channel starting from AY 2021-22.	ERROR
When AY is enter wrong or future year.	EF500035	Please select correct AY. You cannot get prefill data for future Assessment Year.	ERROR
PAN and Aadhaar is not linked	EF00099	Your PAN and Aadhaar is not linked.	ERROR
When user does not enter transactionId.	EF500036	Request cannot be processed as Transaction Id is Null.	ERROR
When ERI client linkage fails	EF500058	The PAN is not a valid client for this ERI.	ERROR
When attributes are incorrect in json request data	EF20123	Invalid Request Data	ERROR

When user tries to generate e-filing OTP more than 5 times	EF00152	You have exceeded the limit to receive OTP. Please try again in 8 hours.	ERROR
When user enters wrong e-filing OTP for more than 3 times	EF00153	You have exceeded the Number of attempts to enter Correct OTP.	ERROR