



DEPARTMENT OF INFORMATION
TECHNOLOGY & COMMUNICATION
(GOVT. OF RAJASTHAN)

DOCUMENT VERIFICATION API'S SPECIFICATION

REPRESENTED BY:

DEPARTMENT OF INFORMATION TECHNOLOGY & COMMUNICATION,
JAIPUR

Document Verification API's Specification

Table of Contents:

1.	Introduction:	2
2.	Purpose of the Document:.....	2
3.	Understanding of All Document Verification Wrapper API's Services:.....	2
3.1.	Token Generate Authentication:.....	2
3.2.	Driving License:.....	4
3.3.	Voter ID card:	9
3.5.	Aadhaar Card:.....	15
3.6.	Jan Aadhaar (Bhamashah):.....	18
3.7.	PAN Card:	22
3.8.	10 th Mark sheet:	25
3.9.	12 th Mark Sheet:	29
3.10.	Police Character Certificate:.....	32
3.11.	Medical Dairy:.....	34
3.12.	BPL Certificate:	37
3.13.	Birth Certificate:	41
3.14.	Death Certificate:	44
3.15.	Marriage Certificate:.....	46
3.16.	Bonafide Certificate:.....	49
3.17.	Labour Card:	52
3.18.	Caste Certificate:	55
3.19.	Registration Certificate (Vehicle):.....	57
3.20.	Disability Certificate:.....	60
3.21.	CBSC Marksheets:	63
3.22.	JamaBandi Khata:	67
3.23.	Jamabandi ROR:.....	70
4.	Log's Detail (Wrapper API's):	73
5.	Error Codes:.....	73
6.	Contact Details for Integration Support:.....	75

Note: - 75

Document Verification API's Specification

1. Introduction:

The Department of Information Technology & Communication, Rajasthan has been created, with the mandate of providing the all type of document verification to the residents. The DoIT&C (Rajasthan) provides online authentication to verify the documents of the document holder.

“Authentication” means the process wherein document type id, document number along with all attributes are submitted for verification on the basis of information, or data or documents available with it. DoIT&C provides an online service to support this process.

2. Purpose of the Document:

According to DoIT&C, this document briefly described the API services that can be accessed through Raj Sevadwar to verify the documents from the departments.

3. Understanding of All Document Verification Wrapper API's Services:

3.1. Token Generate Authentication:

Private URL: <https://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/token>

Request Parameters:

UserName : Enter your SSO ID
Password : WRAPPERAPIUSER
Grant_Type: password

Response json:

Successful Response:

```
{  
    "access_token": "q2W6Gc6qSw5NY2WKOQ-  
8seWnEetShLzkMIW1X1zyveXhts8tUHQdNR_tEIP57Ollwi37BRqXtatewxxF0HyeBMq  
Wvdwwe2dSyarhXLz5ejqHLG3JH8pWZfiXWwT9Bi5xfL6v097KFbO5bXH0myxW7H29FI  
Z2S4qsuRTfLemvDNipdILV2ATcRFiViqlIUt1tTQJ0jqVT-  
pJ6RjeFGX3YQd_Da_Vnd8Aqm8Jt73UZ4k-  
YtDODEqTgGtPpNa22IxT6fwuvHuyRIwU4aMAUNZkeG_J31nyFGZCQTXLq3wUbb_EhA  
ktJ-d-KnHR2MS_fAFxZlwHpcFt20bs899gDiku09Ty3tXauvv-OIDWzwW7Fq8",  
    "token_type": "bearer",  
    "expires_in": 345599  
}
```

Failure Response:

```
{  
    "error": "invalid_grant",  
    "error_description": "Invalid Username or password"  
}
```

Document Verification API's Specification

Description:

Request Parameters:

Parameter Name	Data Type	Description
User Name	String	This parameter indicates the User Name and It will be a string type
Password	String	This parameter indicates the Password and It will be a string type
Grant_Type	String	This parameter indicates the grant type and it will be a static password. It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
access_token	String	This parameter is used for authenticate and it will be a string type
token_type	String	It indicates the type of token
expires_in	String	It shows the expires of the token

Failure Response:

Parameter Name	Data Type	Description
error	String	This parameter indicates the error and it will be a string type
error_description	String	This parameter indicates the error description and it will be a string type

Document Verification API's Specification

3.2. Driving License:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber: Enter your Certificate Number,

DocumentTypeId: 11,

DOB: Enter your Date of Birth

SSOID: Enter your SSOID

ProjectCode: Enter your Project Code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "Driving License",  
    "Data": "{\"?xml\":{\"@version\":\"1.0\",\"@encoding\":\"UTF-  
8\"},\"@standalone\":\"yes\"},\"responseList\":[\"dldetobj\":{\"biolImgObj\":{\"bdDevId\":\"  
0\"},\"biApplno\":\"1283840120\"},\"biBioCapturedDt\":\"2020-08-  
31T00:00:00+05:30\"},\"biBioid\":\"140665NAS*B*NAS*A\",\"biDigest\":\"ZTNiMGM0NDI5  
OGZjMWMxNDlhZmJmNGM4OTk2ZmI5MjQyN2FINDFINDY0OWI5MzRjYTQ5NTk5MWI3ODU  
yYjg1NQ==\",\"biEndorsedt\":\"2020-09-  
03T15:42:05.521+05:30\"},\"biEndorseTime\":\"1970-01-  
01T15:42:05.521+05:30\"},\"biPhoto\":{\"Image\", \"bitokenId\":\"0\", \"biusid\":\"25703\"},\"  
bioObj\":{\"aadharAuthenticated\":\"false\", \"bioApplno\":\"1283840120\"},\"bioBioid\":\"  
140665NAS*B*NAS*A\", \"bioBloodGroup\":\"O+\", \"bioBloodGroupName\":\"O+  
\", \"bioCitiZen\":\"IND\", \"bioDependentRelation\":\"F\", \"bioDlno\":\"AP26 19860001659  
\", \"bioDob\":\"1965-06-14T00:00:00+05:30\"},\"bioEndorsedt\":\"2020-09-  
03T00:00:00+05:30\"},\"bioEndorsementNo\":\"KA01  
/DLR/0008747/2020\"},\"bioEndorseTime\":\"1970-01-  
01T15:42:05.521+05:30\"},\"bioFirstName\":\"SRI VENKATESWARA\", \"bioFullName\":\"SRI  
VENKATESWARA RAO S\", \"bioGender\":\"1\", \"bioGenderDesc\":\"Male  
\", \"bioLastName\":\"S\", \"bioMiddleName\":\"RAO\", \"bioMobileNo\":\"9448862991\", \"  
bioNatName\":\"SRI VENKATESWARA RAO  
S\", \"bioOrganDonor\":\"78\", \"bioPermAdd1\":\"FLAT NO-102, SRI SAI SHRUSHTI  
APTS\", \"bioPermAdd2\":\"NO-39, 19TH CROSS, SIR MV  
NAGAR\", \"bioPermAdd3\":\"BANGALORE  
EAST,BANGALORE\", \"bioPermDistCd\":\"572\", \"bioPermDistName\":\"Bangalore\", \"bioP  
ermPin\":\"560016\", \"bioPermSdName\":\"Bangalore  
East\", \"bioPermSdcode\":\"5544\", \"bioPermVillTownCd\":\"0\", \"bioQmQualcd\":\"9\", \"  
bioRecGenesis\":\"87\", \"bioSwdFname\":\"NARAYANA  
RAO\", \"bioSwdFullName\":\"NARAYANA RAO S  
B\", \"bioSwdLname\":\"B\", \"bioSwdMname\":\"S\", \"bioTempAdd1\":\"FLAT NO-102, SRI
```

Document Verification API's Specification

SAI SHRUSHTI APTS\", \"bioTempAdd2\": \"NO-39, 19TH CROSS, SIR MV NAGAR\", \"bioTempAdd3\": \"BANGALORE EAST,BANGALORE\", \"bioTempDistCd\": \"572\", \"bioTempDistName\": \"Bangalore\", \"bioTempPin\": \"560016\", \"bioTempSdName\": \"Bangalore East\", \"bioTempSdcode\": \"5544\", \"bioTempVillTownCd\": \"0\", \"bioTokenId\": \"0\", \"bioUserId\": \"25703\", \"dbLoc\": \"Applive\", \"dlcovs\": [{\"covIssueAuthCode\": \"KA04\", \"covabbrv\": \"LMV\", \"covdesc\": \"LIGHT MOTOR VEHICLE\", \"dcApplno\": \"9626015\", \"dcAuthDt\": \"2000-09-19T00:00:00+05:30\", \"dcAuthNo\": \"KA04\", \"dcCovcd\": \"4\", \"dcEndorseNo\": \"null\", \"dcEndorsedt\": \"2000-09-19T00:00:00+05:30\", \"dcIssuedt\": \"2000-09-19T00:00:00+05:30\", \"dcLicno\": \"AP26 19860001659\", \"endouserid\": \"0\", \"olaName\": \"MVI,KAVALI,NELLORE DIST\", \"olacd\": \"AP26\", \"veShortdesc\": \"XXXXX\", \"vecatg\": \"NT\"}, {\"covIssueAuthCode\": \"AP26\", \"covabbrv\": \"MCWG\", \"covdesc\": \"Motor Cycle with Gear(Non Transport)\", \"dcApplno\": \"9626015\", \"dcAuthDt\": \"1986-06-13T00:00:00+05:30\", \"dcAuthNo\": \"AP26\", \"dcCovcd\": \"3\", \"dcEndorseNo\": \"null\", \"dcEndorsedt\": \"1986-06-13T00:00:00+05:30\", \"dcIssuedt\": \"1986-06-13T00:00:00+05:30\", \"dcLicno\": \"AP26 19860001659\", \"endouserid\": \"0\", \"olaName\": \"MVI,KAVALI,NELLORE DIST\", \"olacd\": \"AP26\", \"veShortdesc\": \"XXXXX\", \"vecatg\": \"NT\"}], \"dlobj\": {\"boid\": \"140665NAS*B*NAS*A\", \"dlApplno\": \"1283840120\", \"dlDigest\": \"OGI2ZmVjZTZkOTQ4NGM5YTU3MmQyNGJhMWE3MGY5NmU4ZDUyMWQwY2MxYTQ5MGM2NDkyYTdkOTUwNGRmZTdiMQ==\", \"dlDispatchStatus\": \"78\", \"dlEndorseAuth\": \"KA01\", \"dlEndorsedt\": \"2020-09-03T00:00:00+05:30\", \"dlEndoseno\": \"KA01\", \"/DLR/0008747/2020\", \"dlEndorsetime\": \"1970-01-01T15:42:05.521+05:30\", \"dlIncChallanNo\": \"\", \"dlIncRtoAction\": \"\", \"dlIncSourceType\": \"\", \"dlIntermediateStage\": \"\", \"dlIssueauth\": \"AP26\", \"dlIssuedt\": \"1986-06-13T00:00:00+05:30\", \"dlLatestTrcode\": \"514\", \"dlLicno\": \"AP26 19860001659\", \"dlNtValdfrDt\": \"2020-09-03T00:00:00+05:30\", \"dlNtValdtoDt\": \"2025-09-02T00:00:00+05:30\", \"dlPrintDate\": \"2020-09-03T00:00:00+05:30\", \"dlPrintStatus\": \"65\", \"dlRecGenesis\": \"87\", \"dlRtoCode\": \"KA01\", \"dlSeqno\": \"DRAAMYL20\", \"dlStatus\": \"Active\", \"dlTokenId\": \"0\", \"dlUsid\": \"25703\", \"enforceRemark\": \"\", \"olaName\": \"MVI,KAVALI,NELLORE DIST\", \"olacode\": \"AP26\", \"omOfficeTownname\": \"RTO BANGALORE (CENTRAL), HSR LAYOUT\", \"omRtofullname\": \"RTO BANGALORE (CENTRAL), HSR LAYOUT\", \"omRtoShortname\": \"\", \"stateName\": \"Karnataka\", \"statecd\": \"KA\"}, \"errordc\": \"0\", \"transReqObj\": {\"olaCode\": \"KA01\", \"olaOffName\": \"RTO BANGALORE (CENTRAL), HSR LAYOUT\", \"trAbbr\": \"RENEWAL OF DL\", \"trApplno\": \"0\", \"trCovcd\": \"0\", \"trEntrydt\": \"2020-09-03T00:00:00+05:30\", \"trName\": \"RENEWAL OF DL\", \"trSINo\": \"0\", \"trTrCd\": \"0\", \"trVisitNo\": \"0\"}}}}\"}

Failure Response:

```
{  
  "Message": "Data Not Found",  
  "IsSuccess": false,
```

Document Verification API's Specification

"TotalRecordsCount": 0,

Document Verification API's Specification

```
"DocumentName": "Driving License",
"Data": null
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
DOB	String	This parameter indicates the date of birth and It will be a string type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

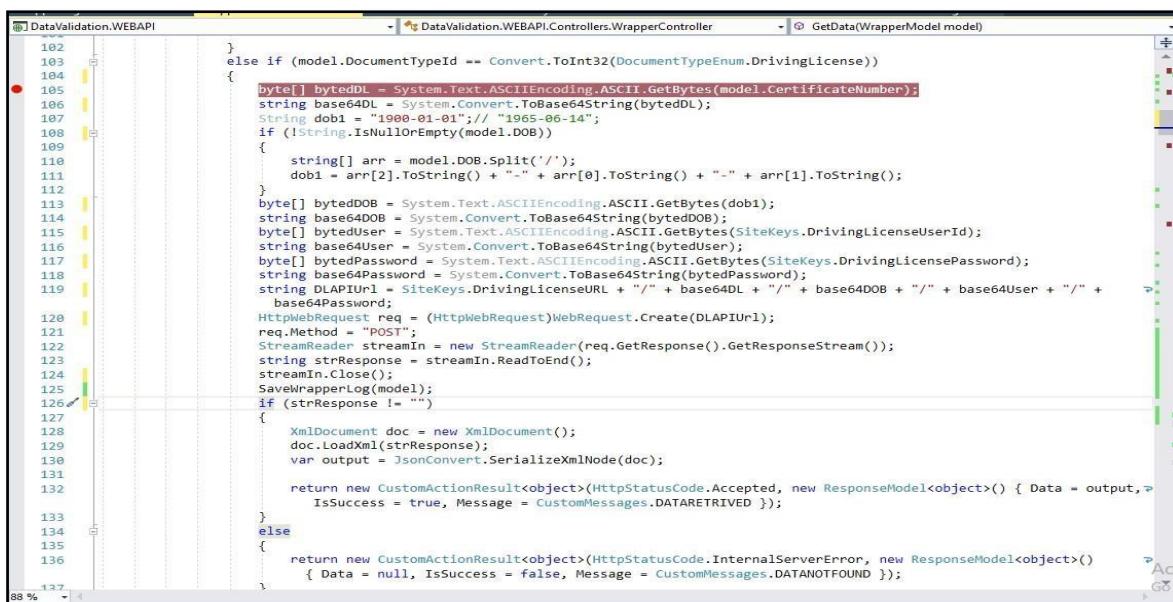
Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string type

Document Verification API's Specification

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type.

Code Snippet:



```

102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
}
else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.DrivingLicense))
{
    byte[] bytedDL = System.Text.Encoding.ASCII.GetBytes(model.CertificateNumber);
    string base64DL = System.Convert.ToBase64String(bytedDL);
    String dob1 = "1900-01-01"; // "1965-06-14";
    if (!String.IsNullOrEmpty(model.DOB))
    {
        string[] arr = model.DOB.Split('/');
        dob1 = arr[2].ToString() + "-" + arr[0].ToString() + "-" + arr[1].ToString();
    }
    byte[] bytedDOB = System.Text.Encoding.ASCII.GetBytes(dob1);
    string base64DOB = System.Convert.ToBase64String(bytedDOB);
    byte[] bytedUser = System.Text.Encoding.ASCII.GetBytes(SiteKeys.DrivingLicenseUserId);
    string base64User = System.Convert.ToBase64String(bytedUser);
    byte[] bytedPassword = System.Text.Encoding.ASCII.GetBytes(SiteKeys.DrivingLicensePassword);
    string base64Password = System.Convert.ToBase64String(bytedPassword);
    string DLAPIUrl = SiteKeys.DrivingLicenseURL + "/" + base64DL + "/" + base64DOB + "/" + base64User + "/" +
        base64Password;
    HttpWebRequest req = (HttpWebRequest)WebRequest.Create(DLAPIUrl);
    req.Method = "POST";
    StreamReader streamIn = new StreamReader(req.GetResponse().GetResponseStream());
    string strResponse = streamIn.ReadToEnd();
    streamIn.Close();
    SaveWrapperLog(model);
    if (strResponse != "")
    {
        XmlDocument doc = new XmlDocument();
        doc.LoadXml(strResponse);
        var output = JsonConvert.SerializeXmlNode(doc);

        return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output, IsSuccess = true, Message = CustomMessages.DATARETRIEVED });
    }
    else
    {
        return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.DATANOTFOUND });
    }
}
}

```

Document Verification API's Specification

3.3. Voter ID card:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeId : 10

SSOID : Enter your SSOID

ProjectCode : Enter your project code

Response Json:

Successfully Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "Voter ID Card",  
    "Data": {  
        "REGISNUMBER": null,  
        "EventDate": null,  
        "Name": null,  
        "FatherName": null,  
        "MohterName": null,  
        "Address": null,  
        "Sex": "Male",  
        "Status": null,  
        "Remarks": null,  
        "NAME": "गगन",  
        "AC_NO": "68",  
        "AGE": "29",  
        "DOB": null,  
        "ADDRESS": "ब्राह्मण मोहWo, पाटन, पाटन, रैणी, अलवर, PINCODE-301414",  
        "DATE_OF_ISSUE": null,  
        "PART_NO": "15",  
        "ISAVAILABLE": null,  
        "RELATION_TYPE": "Father",  
        "RELATION": "गगराज",  
        "EPIC_NO": "NCN0885970",  
        "SL_NO": null,  
        "GENDER": null,  
        "PHOTO": "Yes"  
    }  
}
```

Failure Response:

```
{
```

Document Verification API's Specification

```

    "Message": "Data Not Found",
    "IsSuccess": false,
    "TotalRecordsCount": 0,
    "DocumentName": "Voter ID Card",
    "Data": null
}
  
```

Description:**Request Parameters:**

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be an int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string type

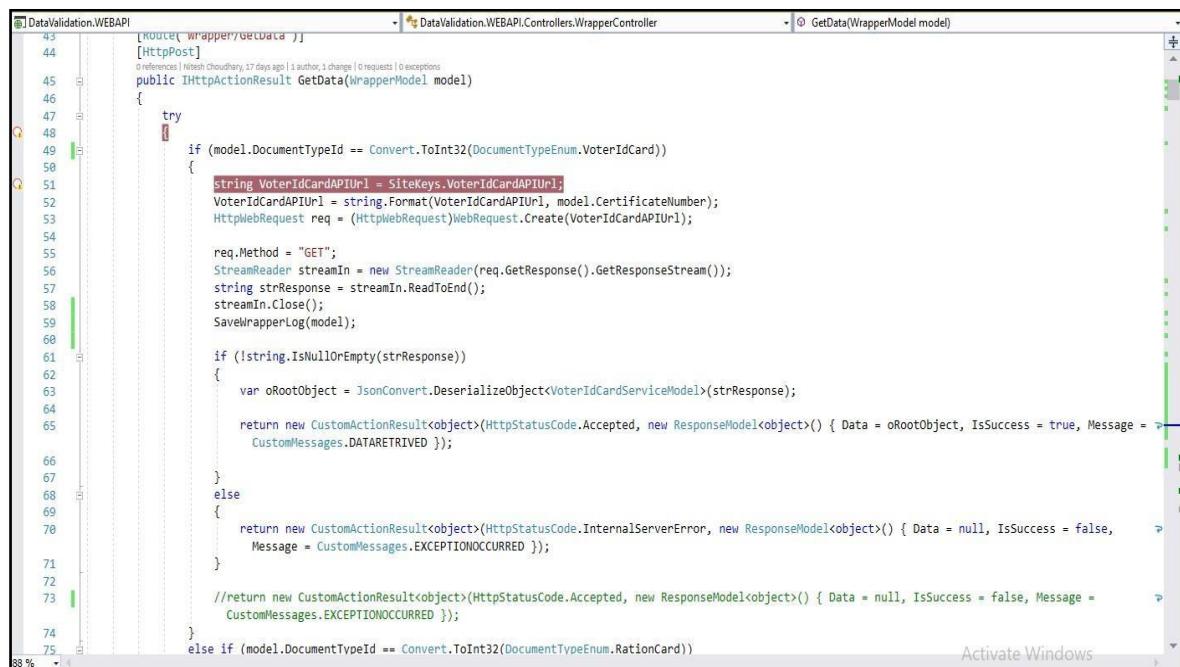
Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string

Document Verification API's Specification

		type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data		This parameter indicates the null data and it will be an string type.

Code Snippet:



The screenshot shows a code editor window with the following details:

- Title Bar:** DataValidation.WEBAPI - DataValidation.WEBAPI.Controllers.WrapperController - GetData(WrapperModel model)
- Code Area:**

```

43 [Route("Wrapper/GetData")]
44 [HttpPost]
45 public IHttpActionResult GetData(WrapperModel model)
46 {
47     try
48     {
49         if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.VoterIdCard))
50         {
51             string VoterIdCardAPIUrl = SiteKeys.VoterIdCardAPIUrl;
52             VoterIdCardAPIUrl = string.Format(VoterIdCardAPIUrl, model.CertificateNumber);
53             HttpWebRequest req = (HttpWebRequest)WebRequest.Create(VoterIdCardAPIUrl);
54
55             req.Method = "GET";
56             StreamReader streamIn = new StreamReader(req.GetResponse().GetResponseStream());
57             string strResponse = streamIn.ReadToEnd();
58             streamIn.Close();
59             SaveWrapperLog(model);
60
61             if (!string.IsNullOrEmpty(strResponse))
62             {
63                 var oRootObject = JsonConvert.DeserializeObject<VoterIdCardServiceModel>(strResponse);
64
65                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = oRootObject, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
66             }
67             else
68             {
69                 return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
70             }
71
72             //return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
73         }
74     }
75 }
```
- Status Bar:** Activate Windows

Document Verification API's Specification

3.4. Ration Card

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeID : 3

SSOID :Enter your SSOID

ProjectCode : Enter your Project Code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrieved",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "Ration Card Verification Document",  
    "Data": "{\"?xml\":{\"@version\":\"1.0\",\"@encoding\":\"UTF-  
8\"},\"rationCardDetail\":{\"flag\":\"true\",\"message\":\"OK\"},\"detail\":{\"fpsDetail\":{\"f  
psCode\":\"12560\"},\"fpsOwnerName\":\"SANTOS KUMAR 6 TO  
11\",\"fpsContactNumber\":\"9694789381\",\"fpsAddress\":\"सांगासीं\",\"fpsBlock\":\"Na  
w algarh\",\"fpsDistrict\":\"Jhunjhunun\"},\"rcDetail\":{\"categoryName\":\"APL\",\"nfsa\":\"  
N/A\",\"criteria\":\"RURAL\"},\"houseNameNo\":\"वारू टो न.टोटा -  
07\"},\"colonyStreet\":\"सांगासीं\",\"tehsil\":\"सांगासीं,  
नवलगढ़\"},\"district\":\"झाझांआन\"  
,\"pin\":\"333705\"},\"state\":\"RAJASTHAN\"},\"rationCardNu  
mber\":\"200003828175\"},\"rationCardName\":\"NARESH KULAHARI/नरेश  
क  
ल्हरातोी\"},\"gasConnection\":\"N\",\"familyDetail\":{\"memberDetail\":[{\"aadharID\":null,  
\"  
contactNumber\":\"9667644472\",\"bhamashahID\":null,\"runningNumber\":\"1\",\"memb  
erNameEN\":\"NARESH KULAHARI/नरेश क  
ल्हरातोी\"},\"memberNameHI\":\"नरेश  
कल्हरातोी\"},\"memberRelationEN\":\"SELF\",\"memberRelationHI\":\"स्वयंत्रता  
\",\"gender\":\"M\",  
\"age\":\"25\",\"memberuid\":\"200003828175001\"},{\"aadharID\":null,\"contactNumber  
\":\"9667644472\",\"bhamashahID\":null,\"runningNumber\":\"2\",\"memberNameEN\":\"  
KAVITA DEVI/कविता देवी\"},\"memberNameHI\":\"कविता  
देवी\"},\"memberRelationEN\":\"WIFE\",\"memberRelationHI\":\"पती\",\"gender\":\"F\",\"a  
ge\":\"28\",\"memberuid\":\"200003828175002\"},{\"aadharID\":null,\"contactNumber\":\"  
9667644472\",\"bhamashahID\":null,\"runningNumber\":\"3\",\"memberNameEN\":\"KRT  
IKA CHODHURI/क्रगतका चोधरी\"},\"memberNameHI\":\"क्रगतका  
चोधरी\"},\"memberRelationEN\":\"DAUGHTER\",\"memberRelationHI\":\"बेटी\",\"gender\":\"  
F\",\"age\":\"1\",\"memberuid\":\"200003828175003\"}]}]}"}  
}
```

Document Verification API's Specification

Failure Response:

```
{  
    "Message": "Please enter correct ration card (RC) number.",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Ration Card Verification Document",  
    "Data": null  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

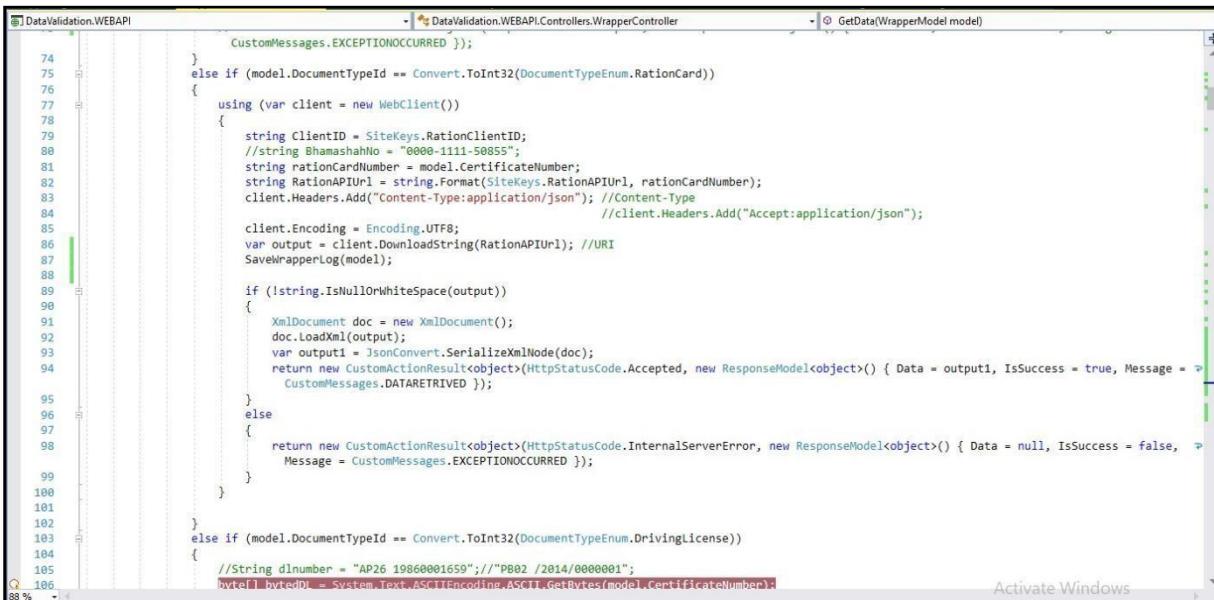
Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string type

Document Verification API's Specification

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type.

Code Snippet:



The screenshot shows a code editor with a C# file named `WrapperController.cs`. The code is part of a project named `DataValidation.WEBAPI`. The specific method shown is `GetData(WrapperModel model)`. The code handles different document types based on their ID. For Ration Card, it uses a WebClient to download XML data from a URL and converts it to JSON. For Driving License, it performs a byte-level comparison between the provided certificate number and a hard-coded value. The code includes various imports like `System.Text`, `System.Linq`, and `System.Xml`.

```
74 } else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.RationCard))
75 {
76     using (var client = new WebClient())
77     {
78         string ClientID = SiteKeys.RationClientID;
79         //string BhamashahNo = "0000-1111-5085";
80         string rationCardNumber = model.CertificateNumber;
81         string RationAPIUrl = string.Format(SiteKeys.RationAPIUrl, rationCardNumber);
82         client.Headers.Add("Content-Type:application/json"); //Content-Type
83         //client.Headers.Add("Accept:application/json");
84
85         client.Encoding = Encoding.UTF8;
86         var output = client.DownloadString(RationAPIUrl); //URI
87         SaveWrapperLog(model);
88
89         if (!string.IsNullOrWhiteSpace(output))
90         {
91             XmlDocument doc = new XmlDocument();
92             doc.LoadXml(output);
93             var output1 = JsonConvert.SerializeXmlNode(doc);
94             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output1, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
95         }
96         else
97         {
98             return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
99         }
100    }
101
102 }
103 else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.DrivingLicense))
104 {
105     //String dinumber = "AP26 19860001659"/"PB02 /2014/0000001";
106     byte[] bytesDl = System.Text.Encoding.ASCII.GetBytes(model.CertificateNumber);
107 }
```

Document Verification API's Specification

3.5. Aadhaar Card:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeID : 6

SSOID : Enter your SSOID

ProjectCode : Enter your Project Code

FullName : Enter your Full Name

Gender : Enter your Gender

DOB: Enter your Date of Birth

MobileNumber: Enter your Mobile Number

Response json:

Successfully Response:

```
{  
    "Message": "Data Retrieved",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "Aadhaar Card",  
    "Data":  
        "{\"$id\": \"1\", \"uidtoken\": \"01002480u2AubbFoGMF9jr0APC5eo3OcRD2IrqU//vzMFUCG  
bWpCkvOFyu35hZdNRsDmHhFT\", \"txn\": \"DOITRJGOVAUTH202201041306124207321\", \"  
status\": \"y\"}"  
}
```

Failure Response:

```
{  
    "Message": "Invalid Aadhaar Number",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Aadhaar Card",  
    "Data":  
        "{\"$id\": \"1\", \"uidtoken\": \"01002480sENt9aj/HChLNAmKBgyYmPpsF0rAFKOiFlgDt2DBOA  
xPTHXMSsCghIyyDndV51F8\", \"txn\": \"DOITRJGOVAUTH202201041307316859906\", \"stat  
us\": \"n\"}"  
}
```

Document Verification API's Specification

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type
Full Name	String	This parameter shows the Full name and it will be a string type
Gender	String	This parameter shows the Gender and it will be a string type
DOB	int	This parameter shows the date of birth and it will be a int type
Mobile Number	int	This parameter shows the Mobile Number and it will be a int type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the

Document Verification API's Specification

		document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type.

Code Snippet:

```

168     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.Aadhaar))
169     {
170         using (var client = new WebClient())
171         {
172             string CertificatePath;
173             string AadhaarNumber = model.CertificateNumber;
174             string FullName = model.FullName;
175             string DOB = model.DOB;
176             string Gender = model.Gender == "Male" ? "M" : model.Gender == "Female" ? "F" : "O";
177             string timeStamp = DateTime.Now.ToString("yyyy-MM-ddHH\\mm\\ss");
178             string msg = "<xml><version>1.0</version><encoding>utf-8</encoding><PId>" + timeStamp + "<ver>2.0</ver><Pi ms='E' name='"
179             + gender + "' gender=" + gender + "' dob=" + DOB + "'></Demo></PId>";
180             string expiryDate = string.Empty;
181             string encSessionKey = string.Empty;
182             string encryptedHmacBytes = string.Empty;
183             string encXMLPIDDataNew = GetEncryptedPIDXmlNew(msg, CertificatePath, timeStamp, ref expiryDate, ref encSessionKey, ref
184             encryptedHmacBytes);
185             string authXML = "<xml version='1.0' encoding='UTF-8'><authrequest uid='"
186             + AadhaarNumber + "' tid='"
187             + subAuaa + "' subuaa='"
188             + SiteKeys.SUBUAADHAR + "' bte='"
189             + SiteKeys.AadhaarAPILicenseKey + "' rcc='"
190             + SiteKeys.mec + "' ver='"
191             + SiteKeys.devicrError + "' ip='"
192             + SiteKeys.ipc + "' fdc='"
193             + SiteKeys.idc + "' macaddr='"
194             + SiteKeys.lot + "' lot='"
195             + SiteKeys.nmPoints + "' nmPoints='"
196             + SiteKeys.fCount + "' nma='"
197             + SiteKeys.errInfoDS + "' errInfoDS='"
198             + SiteKeys.pCount + "' pCount='"
199             + SiteKeys.errCodeRDS + "' errCodeRDS='"
200             + SiteKeys.errInfoRDS + "' errInfoRDS='"
201             + SiteKeys.sKey + "' sKey='"
202             + SiteKeys.expiryDate + "'><Skey cl='"
203             + SiteKeys.sKey + "'></Skey><Hmac>' +"
204             encryptedHmacBytes + "</Hmac><Data type='X'>" + encXMLPIDDataNew + "</Data></authrequest>";
205             var response = postXMLData(SiteKeys.AadhaarAPIUrl, authXML); //Public//Preprod
206             XmlDocument tournamentsXML = new XmlDocument();
207             tournamentsXML.LoadXml(response);
208             XmlDocument doc = new XmlDocument();
209             doc.LoadXml(response);
210             var output = JsonConvert.SerializeXmlNode(doc);
211             if (output.Contains("@"))
212             {
213                 output.Replace("@", "");
214                 myDetails = JsonConvert.DeserializeObject<object>(jsonxmlConvert<object>(output));
215                 string outputobject = JsonConvert.SerializeObject(myDetails.authresponse.auth);
216                 SaveWrapperLog(model);
217                 if (!string.IsNullOrWhiteSpace(outputobject))
218                 {
219                     return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = outputobject, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
220                 }
221                 else
222                 {
223                     new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
224                 }
225             }
226         }
227     }
228 }

```

Document Verification API's Specification

3.6. Jan Aadhaar (Bhamashah):

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeID : 7

SSOID: Enter your SSOID

ProjectCode : Enter your Project Code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "JanAadhaar Card",  
  
    "Data": "[{\\"ENR_ID\\":\\"9999-62FJ-  
00245\\",\\"AADHAR_ID\\":\\"838172193254\\",\\"NAME\\":\\"Manju\\",\\"MOBILE_NO\\":\\"94609  
15879\\",\\"SRDR_MID\\":\\"0\\",\\"JAN_MEMBER_ID\\":\\"54480613692\\",\\"JAN_AADHAR\\":\\"4  
852582609\\",\\"NAME_HINDI\\":\\"મજૂ\\",\\"SRDR_ID\\":\\"VWPJSSR\\"},{\\"ENR_ID\\":\\"9999-  
62FJ-00245\\",\\"AADHAR_ID\\":\\"  
\",\\"NAME\\":\\"Saanvi\\",\\"MOBILE_NO\\":null,\\"SRDR_MID\\":\\"144807218\\",\\"JAN_MEMBE  
R_ID\\":\\"98665237098\\",\\"JAN_AADHAR\\":\\"4852582609\\",\\"NAME_HINDI\\":\\"સાન્વી\\",\\"S  
RDR_ID\\":\\"VWPJSSR\\"},{\\"ENR_ID\\":\\"9999-62FJ-  
00245\\",\\"AADHAR_ID\\":\\"489414935298\\",\\"NAME\\":\\"Tulsa  
Ram\\",\\"MOBILE_NO\\":\\"9460915879\\",\\"SRDR_MID\\":\\"104569847\\",\\"JAN_MEMBER_ID\\"  
":\\"47778110710\\",\\"JAN_AADHAR\\":\\"4852582609\\",\\"NAME_HINDI\\":\\"તલસા  
રામ\\",\\"SRDR_ID\\":\\"VWPJSSR\\"},{\\"ENR_ID\\":\\"9999-62FJ-00245\\",\\"AADHAR_ID\\":\\"  
\",\\"NAME\\":\\"Sakshi  
Parihar\\",\\"MOBILE_NO\\":null,\\"SRDR_MID\\":\\"104569846\\",\\"JAN_MEMBER_ID\\":\\"38021  
468386\\",\\"JAN_AADHAR\\":\\"4852582609\\",\\"NAME_HINDI\\":\\"સાક્ષી  
પરહાર\\",\\"SRDR_ID\\":\\"VWPJSSR\\"}]"  
}
```

Failure Response:

```
{  
    "Message": "Invalid JanAadhaar Number",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "JanAadhaar Card",  
    "Data": "{\"status\":\"false\",\"errMsg\":\"error Cursor is closed.\\"}"  
}
```

Document Verification API's Specification

}

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the

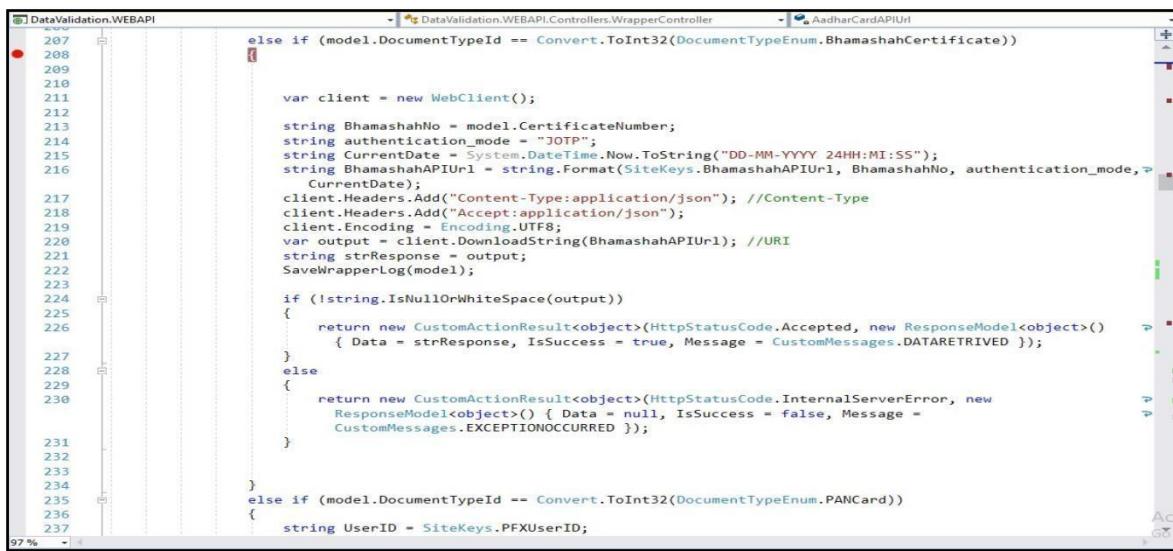
Document Verification API's Specification

	total records count, if total records count will be wrong.
--	--

Document Verification API's Specification

DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



The screenshot shows a code editor with a C# file named `WrapperController.cs`. The code is part of a project named `DataValidation.WEBAPI`. The code handles document verification requests based on the document type. It uses a `WebClient` to make a request to the `BhamashahAPIUrl` and then processes the response. The code includes logic for Bhamashah Certificate and PAN Card verification. The code editor interface shows line numbers from 207 to 237, and the status bar at the bottom indicates 97% completion.

```
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237
```

```
else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.BhamashahCertificate))
{
    var client = new WebClient();

    string BhamashahNo = model.CertificateNumber;
    string authentication_mode = "JOTP";
    string CurrentDate = System.DateTime.Now.ToString("DD-MM-YYYY 24HH:MI:SS");
    string BhamashahAPIUrl = string.Format(SiteKeys.BhamashahAPIUrl, BhamashahNo, authentication_mode, CurrentDate);
    client.Headers.Add("Content-Type:application/json"); //Content-Type
    client.Headers.Add("Accept:application/json");
    client.Encoding = Encoding.UTF8;
    var output = client.DownloadString(BhamashahAPIUrl); //URI
    string strResponse = output;
    SaveWrapperLog(model);

    if (!string.IsNullOrWhiteSpace(output))
    {
        return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>()
        {
            Data = strResponse, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
    }
    else
    {
        return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>()
        {
            Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
    }
}

else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.PANCard))
{
    string UserID = SiteKeys.PFXUserID;
```

Document Verification API's Specification

3.7. PAN Card:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number
DocumentTypeID : 12
SSOID: Enter your SSOID
ProjectCode : Enter your Project Code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "PAN Card",  
    "Data": "{\"Status\":\"Success\",\"PanNo\":\"ASJPA5799B\",\"PanStatus\":\"EXISTING AND  
VALID\",\"LastName\":\"ASWANI\",\"FirstName\":\"ASHISH\",\"MiddleName\":\"\",\"PanTitle\":\"Shri\",\"LastUpda  
teDate\":\"22/05/2010\",\"NameOnCard\":\"\",\"Filter1\":\"\",\"Filter2\":\"\"}"  
}
```

Failure Response:

```
{  
    "Message": "Record (PAN) Not Found in ITD Database/Invalid PAN",  
    "IsSuccess": true,  
    "TotalRecordsCount": 0,  
    "DocumentName": "PAN Card",  
    "Data": "{\"Status\":\"Success\",\"PanNo\":\"ASJPA5799E\",\"PanStatus\":\"Record (PAN) Not Found in  
ITD Database/Invalid  
PAN\",\"LastName\":\"\",\"FirstName\":\"\",\"MiddleName\":\"\",\"PanTitle\":\"\",\"LastUpdateDate\":\"\",\"NameOnCa  
rd\":\"\",\"Filter1\":\"\",\"Filter2\":\"\"}"  
}
```

Error Code:

Error Code	Error Description
1	Success.
2	System Error.
3	Authentication Failure.
4	User not authorized.
5	No PANs Entered.
6	User validity has expired.
7	Number of PANs exceeds the limit (5).
8	Not enough balance.
9	Not a HTTPs request.
10	Post method not used.

Document Verification API's Specification

12

WRONG_VERSION_FOR_RESPONSE.

Document Verification API's Specification

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type

Document Verification API's Specification

TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:

```

235     else if (model.DocumentTypeID == Convert.ToInt32(DocumentTypeEnum.PANCard))
236     {
237         string UserID = Sitekeys.PFXUserID;
238         string apiml = Sitekeys.PanServiceML;
239         string PFXPassword = Sitekeys.PFXPassword;
240         string CertificateName = Sitekeys.CertificateName;
241         string certificate = model.Certificate;
242         System.Text.UTF8Encoding encoding = new System.Text.UTF8Encoding();
243         X509Certificate2 m = new X509Certificate2(System.Web.HttpContext.Current.Server.MapPath("~/PFX/") + CertificateName, PFXPassword);
244         string certificate2 = certificate;
245         byte[] bytes = encoding.GetBytes(arg);
246         byte[] sig = Sign(bytes, m);
247         string postdata = Encoding.UTF8.GetString(sig);
248         Stringbuilder postData = new Stringbuilder();
249         postData.Append("data=" + Arg);
250         postData.Append("&version=" + 2);
251         postData.Append("&signature=" + System.Web.HttpUtility.UrlEncode(sig));
252         byte[] data = encoding.GetBytes(postData.ToString());
253         ServicePointManager.Expect100Continue = true;
254         HttpWebRequest myRequest = (HttpWebRequest)WebRequest.Create(apiml);
255         myRequest.Method = "POST";
256         myRequest.ContentType = "application/x-www-form-urlencoded";
257         myRequest.ContentLength = data.Length;
258         ServicePointManager.ServerCertificateValidationCallback = new System.Net.Security.RemoteCertificateValidationCallback(AcceptAllCertifications);
259         ServicePointManager.Expect100Continue = true;
260
261         Stream newStream = myRequest.GetRequestStream();
262         myRequest.Method = "POST";
263         myRequest.Host = "pancardapi";
264         myRequest.ContentType = "application/x-www-form-urlencoded";
265         myRequest.ContentLength = data.Length;
266         Stream Answer = myRequest.GetResponse().GetResponseStream();
267         StreamReader AnswerReader = new StreamReader(Answer);
268         string Response = Answer.ReadToEnd();
269         newStream.Close();
270         string arrResponse = Response.Trim().Split('`');
271         string strResponse = arrResponse[0];
272         for (int i = 0; i < arrResponse.Length; i++)
273         {
274             string fieldName = Enum.GetName(typeof(PanFieldName), Convert.ToInt32(i + 1));
275             if (i == 0)
276             {
277                 json += fieldName + ":" + Enum.GetName(typeof(PanCardErrorCode), Convert.ToInt32(arrResponse[i])).ToString() + "`";
278             }
279             else if (i == 2)
280             {
281                 Dictionary<string, string> dictionary = new Dictionary<string, string>();
282                 json += "," + fieldName + ":" + dictionary.GetPanCardDescription(arrResponse[i].ToString()) + "`";
283             }
284             else
285             {
286                 json += "," + fieldName + ":" + arrResponse[i].ToString() + "`";
287             }
288         }
289         json += "`";
290         strResponse = json;
291         SaveInperLog(model);
292         return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = strResponse, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
293     }

```

3.8. 10th Mark sheet:

PrivateURL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

NOTE: If Marksheets (PDF) is required then send IsPdf = true

Request Parameters:

RollNumber: Enter your Roll Number

ExamType: Enter your Exam type

ExamYear: Enter your Exam Year

DocumentTypeID : 29

SSOID: Enter Your SSOID

ProjectCode : Enter your Project Code

IsPdf: false

Response json:

Successful Response:

Document Verification API's Specification

```
{  
    "Message": "Data Retrieved",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "RBSC 10th Marksheets Main",  
    "Data": [  
        {  
            "CNAME": "HASEENA",  
            "FNAME": "SAMDA KATHAT",  
            "MNAME": "PREM",  
            "DocTypeCode": 10,  
            "DocBase64": null,  
            "Exam_YEAR": "2020",  
            "rollNo": "3900014",  
            "ENROL": "20/0005410",  
            "MM": "600",  
            "MT": "246",  
            "RESREM": "THIRD DIVISION",  
            "DOB": "21/04/2005",  
            "Per": 41.0,  
            "Grp": null  
        }  
    ]  
}
```

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "RBSC 10th Marksheets Main",  
    "Data": null  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
Roll Number	int	This parameter indicates the Roll Number of 10 th marksheets and It will be a int type
Exam Type	int	This parameter indicates the Exam type and it will be a int type

Document Verification API's Specification

Exam Year	int	This Parameter indicates the exam year of 10 th and it will be a int type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type
IsPdf	Boolean	This parameter checks the PDF and it will be a Boolean type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string

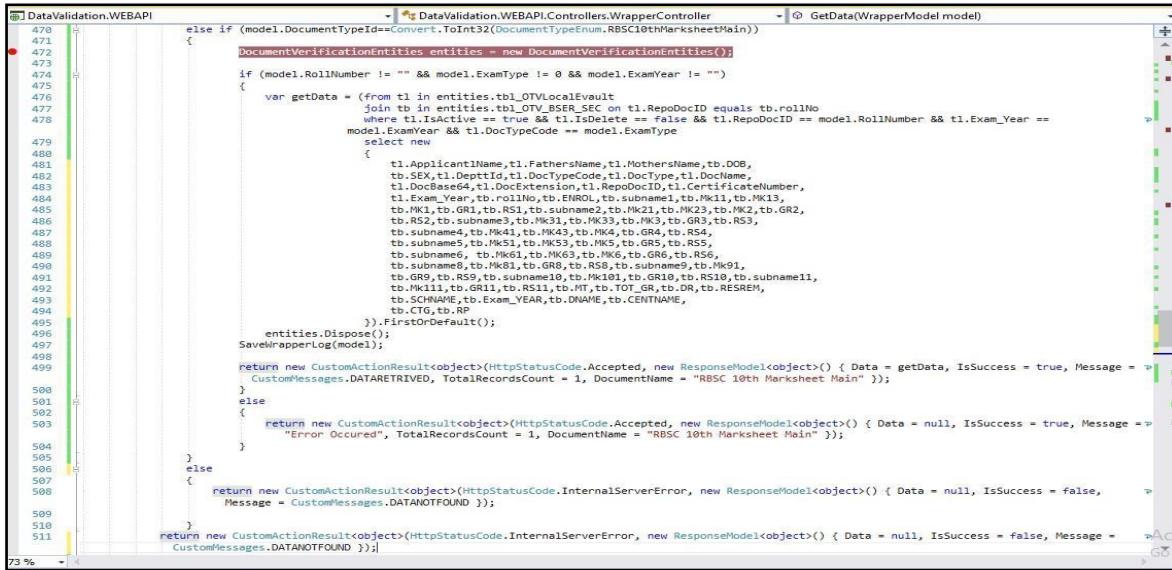
Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.

Document Verification API's Specification

DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



```

470     else if (model.DocumentTypeID == Convert.ToInt32(DocumentTypeEnum.RBSC10thMarksheetMain))
471     {
472         DocumentVerificationEntities entities = new DocumentVerificationEntities();
473
474         if (model.RollNumber != "" && model.ExamType != 0 && model.ExamYear != "")
475         {
476             var getData = (from t1 in entities.tbl_OTV_LocalEVault
477                           join t1_1 in entities.tbl_OTV_BSER_SEC on t1.RepoDocID equals t1_1.RepoDocID
478                           where t1_1.IsActive == true && t1_1.IsDelete == false && t1.RepoDocID == model.RollNumber && t1.Exam_Year ==
479                           model.ExamYear && t1.DocTypeCode == model.ExamType
480                           select new
481                           {
482                               t1.ApplicantName, t1.FatherName, t1.MothersName, tb.DOB,
483                               tb_SEX, tb_Par1ID, tb_DocTypeCode, tb_DocType, t1_1.DocumentID,
484                               t1_1.ExamYear, t1.rollNo, tb.EHROD, tb.subname1, tb.MK11, tb.MK13,
485                               tb.MK1, tb.GR1, tb.RS1, tb.subname2, tb.MK21, tb.MK23, tb.MK2, tb.GR2,
486                               tb.RS2, tb.subname3, tb.MK31, tb.MK33, tb.MK3, tb.GR3, tb.RS3,
487                               tb.subname4, tb.MK41, tb.MK4, tb.MK4, tb.GR4, tb.RS4,
488                               tb.MK51, tb.MK53, tb.MK5, tb.GR5, tb.RS5,
489                               tb.subname6, tb.MK61, tb.MK63, tb.MK6, tb.GR6, tb.RS6,
490                               tb.MK71, tb.GR7, tb.RS7, tb.subname7, tb.MK73,
491                               tb.GR8, tb.RS8, tb.subname8, tb.MK81, tb.GR8, tb.RS8, tb.subname9, tb.Mk91,
492                               tb.GR9, tb.RS9, tb.subname10, tb.MK101, tb.GR10, tb.RS10, tb.subname11,
493                               tb.MK111, tb.GR11, tb.RS11, tb.MT, tb.TOT_GR, tb.DR, tb.RESREM,
494                               tb.SCHMIE, tb.Exam_YEAR, tb.DNAME, tb.CENTNAME,
495                               tb.TG, tb.RP
496                           }).FirstOrDefault();
497
498             entities.Dispose();
499             SaveWrapperLog(model);
500
501             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = getData, IsSuccess = true, Message = "CustomMessages.DATARETRIVED, TotalRecordsCount = 1, DocumentName = \"RBSC 10th Marksheets Main\""});
502         }
503         else
504         {
505             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = true, Message = "Error Occurred", TotalRecordsCount = 1, DocumentName = "RBSC 10th Marksheets Main" });
506         }
507     }
508     else
509     {
510         return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "CustomMessages.DATANOTFOUND" });
511     }
512
513     return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "CustomMessages.DATANOTFOUND" });
514 }

```

Document Verification API's Specification

3.9. 12th Mark Sheet:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

NOTE: If Marksheets (PDF) is required then send parameter IsPdf = true

Request Parameters:

SSOID: Enter your SSOID

DocumentTypeId: 31

RollNumber: Enter your Roll Number

ExamType: Enter your exam type

ExamYear: Enter your Exam Year

ProjectCode : Enter your Project Code

IsPdf: false

Response json:

Successful Response:

```
{  
    "Message": "Data Retrieved",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "RBSC 12th Marksheets Main",  
    "Data": [  
        {  
            "CNAME": "AMAN PATHRA",  
            "FNAME": "MAHENDRA PATHRA",  
            "MNAME": "BEENA PATHRA",  
            "DocTypeCode": 11,  
            "DocBase64": null,  
            "Exam_YEAR": "2020",  
            "rollNo": "2500001.0",  
            "ENROL": null,  
            "MM": "500",  
            "MT": "407",  
            "RESREM": "FIRST DIVISION",  
            "DOB": null,  
            "Per": 81.4,  
            "Grp": "SCIENCE"  
        }  
    ]  
}
```

Failure Response:

```
{
```

Document Verification API's Specification

```

    "Message": "Data Not Found",
    "IsSuccess": false,
    "TotalRecordsCount": 0,
    "DocumentName": "RBSC 12th Marksheets Main",
    "Data": null
}

```

Description:

Request Parameters:

Parameter Name	Data Type	Description
Roll Number	int	This parameter indicates the Roll Number of 12 th marksheets and it will be an int type
Exam Type	int	This parameter indicates the Exam type and it will be an int type
Exam Year	int	This Parameter indicates the exam year of 12 th and it will be an int type
DocumentTypeId	int	This parameter indicates the Document type id and it will be an int type
SSOID	String	This parameter indicates the SSOID and it will be a string type
ProjectCode	String	This parameter indicates the Project code and it will be a string type
IsPdf	Boolean	This parameter checks the PDF and it will be a Boolean type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type

Document Verification API's Specification

DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter indicates the error description and it will be a string type
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:

```

351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
    }
    else if (model.DocumentTypeID == Convert.ToInt32(DocumentTypeEnum.RBSC12thMarksheetMain))
    {
        DocumentVerificationEntities entities = new DocumentVerificationEntities();
        if (model.RollNumber != "" && model.ExamType != 0 && model.ExamYear != "")
        {
            var getData = (from tl in entities.tbl_OTVLocalVault
                          join tb in entities.tbl_OTV_BSER_SR_SEC on tl.RepoDocID equals tb.rollNo
                          where tl.IsActive == true && tl.IsDelete == false && tl.RepoDocID == model.RollNumber && tl.Exam_Year == model.ExamYear && tl.DocTypeCode == model.ExamType
                          select new
                          {
                              tl.Applicant1Name, tl.FathersName, tl.MothersName, tl.DeptId,
                              tl.DocTypeCode, tl.DocType, tl.DocName, tl.DocBase64,
                              tl.DocExtension, tl.RepoDocID, tl.CertificateNumber,
                              tl.Exam_Year, tb.rollNo, tb.subname1, tb.rollRank1,
                              tb.QLF1, tb.subname2, tb.Mark2, tb.QLF2, tb.subname3,
                              tb.QLF3, tb.subname4, tb.Mark4, tb.QLF4, tb.subname5, tb.Mark5,
                              tb.QLF5, tb.subname6, tb.Mark6, tb.QLF6, tb.subname7,
                              tb.QLF7, tb.subname8, tb.Mark8, tb.QLF8, tb.subname9,
                              tb.QLF9, tb.subname10, tb.Mark10, tb.QLF10, tb.subname11,
                              tb.QLF11, tb.subname12, tb.Mark12, tb.QLF12, tb.subname13,
                              tb.QLF13, tb.QLF14, tb.MAX_Marks, tb.GRAND_TOT,
                              tb.PER, tb.RESREM, tb.RESDATE, tb.REG_PVT,
                              tb.district, tb.SCHNAME
                          }).FirstOrDefault();
            entities.Dispose();
            return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>()
            {
                Data = getData, IsSuccess = true, Message = CustomMessages.DATARETRIVED, TotalRecordsCount = 1, DocumentName = "RBSC 12th Marksheets Main"
            });
        }
        else
        {
            return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>()
            {
                Data = null, IsSuccess = true, Message = "Error Occurred", TotalRecordsCount = 1, DocumentName = "RBSC 12th Marksheets Main"
            });
        }
    }
}

```

Document Verification API's Specification

3.10. Police Character Certificate:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

SSOID: Enter your SSOID
DocumentTypeId: 18
CertificateNumber: Enter your Certificate Number
ProjectCode : Enter your project code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "PoliceCharacter Certificate",  
    "Data":  
        "{\"MESSAGE\":\"SUCCESS\",\"STATUS_CODE\":\"200\",\"Police_Verification_Token_Number\":1652663,\"URL\":\"https://emitraapp.rajasthan.gov.in/emitraApps/printCertificatePage?urlOfPrinting=https://emitraapp.rajasthan.gov.in//emitrashared/uploadeddocs/tempCert/1641287251041/1652663_PoliceVerification.pdf\"}"}  
}
```

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "PoliceCharacter Certificate",  
    "Data": null  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type

Document Verification API's Specification

SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

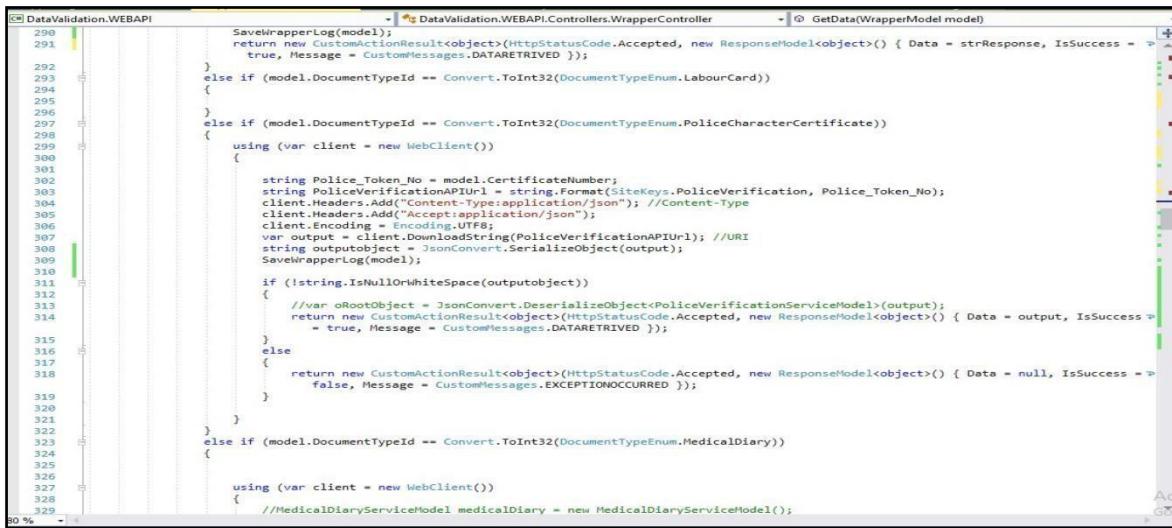
Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Code Snippet:



```

298     SaveWrapperLog(model);
299     return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = strResponse, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
300   }
301   else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.LabourCard))
302   {
303   }
304   else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.PoliceCharacterCertificate))
305   {
306     using (var client = new WebClient())
307     {
308       string Police_Token_No = model.CertificateNumber;
309       string PoliceVerificationAPIUrl = string.Format(SiteKeys.PoliceVerification, Police_Token_No);
310       client.Headers.Add("Content-Type:application/json"); //Content-Type
311       client.Headers.Add("Accept:application/json");
312       client.Encoding = Encoding.UTF8;
313       var output = client.DownloadString(PoliceVerificationAPIUrl); //URI
314       string outputobject = JsonConvert.SerializeObject(output);
315       SaveWrapperLog(model);
316       if (!string.IsNullOrEmpty(outputobject))
317       {
318         //var oRootObject = JsonConvert.DeserializeObject<PoliceVerificationServiceModel>(output);
319         //return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
320       }
321     }
322   }
323   else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.MedicalDiary))
324   {
325   }
326   else
327   {
328     using (var client = new WebClient())
329     {
330       //MedicalDiaryServiceModel medicalDiary = new MedicalDiaryServiceModel();
331     }
332   }
333 }
334 }
```

3.11. Medical Dairy:

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

SSOID: Enter your SSOID

DocumentTypId: 19

CertificateNumber: Enter your Certificate Number

ProjectCode : Enter your project code

Response json:

Successful Response:

```
{
  "Message": "Data Retrieved",
  "IsSuccess": true,
  "TotalRecordsCount": 1,
  "Data":
    "{\"MESSAGE\":\"SUCCESS\",\"STATUS_CODE\":\"200\",\"URL\":\"https://emitraapp.rajasthan.gov.in/emitraApps/printCertificatePage?urlOfPrinting=https://emitraapp.rajasthan.gov.in//emitrashared/uploadeddocs/tempCert/Pension_Medical_Diary_1627633785794.pdf\"}",
  "DocumentName": "Medical Dairy Certificate",
}
```

Failure Response:

```
{
  "Message": "Data Not Found",
  "IsSuccess": false,
  "TotalRecordsCount": 0,
  "DocumentName": "Medical Dairy Certificate",
```

Document Verification API's Specification

```

    "Data": null
}

```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter show all the API data and it will be a string

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the

Document Verification API's Specification

		error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



```

 318     return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = true, Message = CustomMessages.DATARETRIEVED });
 319 }
 320 }
 321 }
 322 }
 323 else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.MedicalDiary))
 324 {
 325
 326     using (var client = new WebClient())
 327     {
 328         //MedicalDiaryServiceModel medicalDiary = new MedicalDiaryServiceModel();
 329         string Pension_Dept_Token = model.CertificateNumber;
 330         string MedicalDiaryAPIUrl = string.Format(sitekeys.MedicalDiaryAPIUrl, Pension_Dept_Token);
 331         client.Headers.Add("Content-Type:application/json"); //Content-Type
 332         client.Headers.Add("Accept:application/json");
 333         client.Encoding = Encoding.UTF8;
 334         string output = client.DownloadString(MedicalDiaryAPIUrl); //URI
 335         string outputObject = JsonConvert.SerializeObject(output);
 336         SaveWrapperLog(model);
 337
 338         if (!string.IsNullOrWhiteSpace(outputObject))
 339         {
 340             //var oRootObject = JsonConvert.DeserializeObject<PoliceVerificationServiceModel>(output);
 341             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = outputObject, IsSuccess = true, Message = CustomMessages.DATARETRIEVED });
 342         }
 343         else
 344         {
 345             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = true, Message = CustomMessages.EXCEPTIONOCCURRED });
 346         }
 347
 348     }
 349 }
 350 }
 351 else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.RBSC12thMarksheetMain))
 352 {
 353     using (var client = new WebClient())
 354     {
 355         RBSC12thMarksheetMainServiceModel rbSC12thMarksheetMainServiceModel = new RBSC12thMarksheetMainServiceModel();
 356
 357         string output = client.DownloadString(rbSC12thMarksheetMainServiceModel.URL);
 358
 359         if (!string.IsNullOrWhiteSpace(output))
 360         {
 361             //var oRootObject = JsonConvert.DeserializeObject<RBSC12thMarksheetMain>(output);
 362             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output, IsSuccess = true, Message = CustomMessages.DATARETRIEVED });
 363         }
 364         else
 365         {
 366             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = true, Message = CustomMessages.EXCEPTIONOCCURRED });
 367         }
 368     }
 369 }
 370 else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.DisabilityCertificate))
 371 {
 372     using (var client = new WebClient())
 373     {
 374         DisabilityCertificateServiceModel disabilityCertificateServiceModel = new DisabilityCertificateServiceModel();
 375
 376         string output = client.DownloadString(disabilityCertificateServiceModel.URL);
 377
 378         if (!string.IsNullOrWhiteSpace(output))
 379         {
 380             //var oRootObject = JsonConvert.DeserializeObject<DisabilityCertificate>(output);
 381             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output, IsSuccess = true, Message = CustomMessages.DATARETRIEVED });
 382         }
 383         else
 384         {
 385             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = true, Message = CustomMessages.EXCEPTIONOCCURRED });
 386         }
 387     }
 388 }
 389 }
 390 }
 391 }
 392 }
```

Document Verification API's Specification

3.12. BPL Certificate:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

SSOID: Enter your SSOID

DocumentTypeID: 14

CertificateNumber: Enter your Certificate Number

ProjectCode : Enter your Project code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "Data": "{\"?xml\":{\"@version\":\"1.0\", \"@encoding\":\"UTF-  
8\"},\"rationCardDetail\":{\"flag\":\"true\",\"message\":\"OK\",\"detail\":{\"fpsDetail\":{\"f  
psCode\":\"12560\",\"fpsOwnerName\":\"SANTOS KUMAR 6 TO  
11\",\"fpsContactNumber\":\"9694789381\",\"fpsAddress\":\"सांगासी\\\"fpsBlock\":\"Na  
w algarh\\\"fpsDistrict\":\"Jhunjhunun\"},\"rcDetail\":{\"categoryName\":\"APL\",\"nfsa\":\"  
N/A\",\"criteria\":\"RURAL\",\"houseNameNo\":\"वारुंटो नं.टोटा -  
07\",\"colonyStreet\":\"सांगासी\\\"tehsil\":\"सांगासी,  
नवलगढ\\\"district\":\"झांता झांता\\\"pin\":\"333705\\\"state\":\"RAJASTHAN\\\"rationCardNu  
mber\":\"200003828175\\\"rationCardName\":\"NARESH KULAHARI/नरेश  
क  
लहरा\\\"gasConnection\":\"N\\\"familyDetail\":{\"memberDetail\":[{\"aadharID\":null,  
\"  
contactNumber\":\"9667644472\\\"bhamashahID\":null,\"runningNumber\":\"1\\\"memb  
erNameEN\":\"NARESH KULAHARI/नरेश क  
लहरा\\\"memberNameHI\":\"नरेश  
कलहरा\\\"memberRelationEN\":\"SELF\\\"memberRelationHI\":\"स्वयंटोटा  
\\\"gender\":\"M\\\"age\":\"25\\\"memberuid\":\"200003828175001\"},{\"aadharID\":null,  
\"contactNumber  
\":\"9667644472\\\"bhamashahID\":null,\"runningNumber\":\"2\\\"memberNameEN\":\"KA  
VITA DEVI/कववता देवी\\\"memberNameHI\":\"कववता  
देवी\\\"memberRelationEN\":\"WIFE\\\"memberRelationHI\":\"पती\\\"gender\":\"F\\\"a  
ge\":\"28\\\"memberuid\":\"200003828175002\"},{\"aadharID\":null,\"contactNumber\\\"  
\"9667644472\\\"bhamashahID\":null,\"runningNumber\":\"3\\\"memberNameEN\":\"KRT  
IKA CHODHURI/क्रगतका चोधरी\\\"memberNameHI\":\"क्रगतका  
चोधरी\\\"memberRelationEN\":\"DAUGHTER\\\"memberRelationHI\":\"बेटी\\\"gender\":\"  
F\\\"age\":\"1\\\"memberuid\":\"200003828175003\\\"}]}]}},  
37
```

Document Verification API's Specification

```

    "DocumentName": "BPL Certificate",
}

```

Failure Response:

```

{
  "Message": "Exception Occurred",
  "IsSuccess": false,
  "TotalRecordsCount": 0,
  "DocumentName": "BPL Certificate",
  "Data": null
}

```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an

Document Verification API's Specification

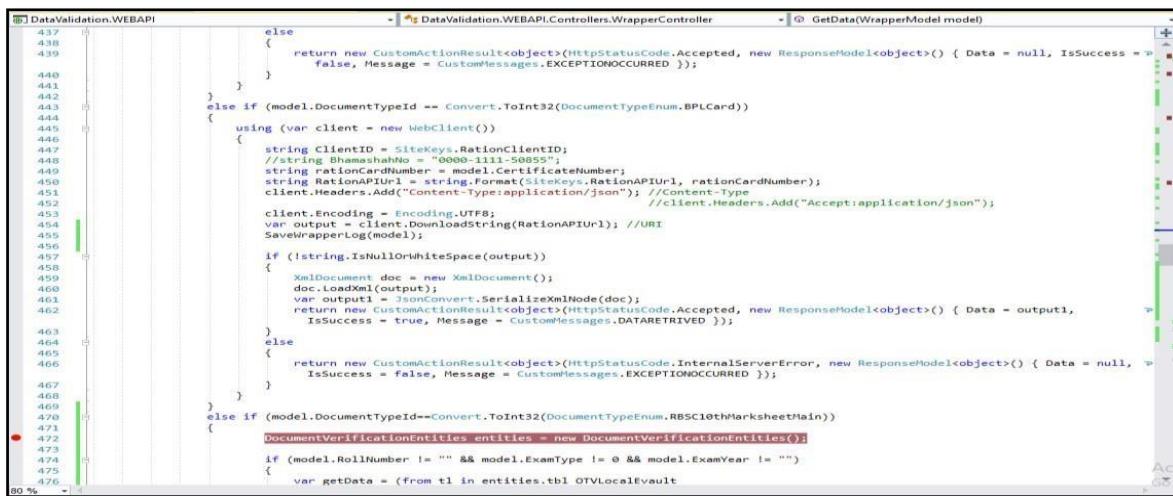
	string type
--	-------------

Document Verification API's Specification

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



```

437     else
438     {
439         return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTION_OCCURRED });
440     }
441 }
442 else if (model.DocumentTypeID == Convert.ToInt32(DocumentTypeEnum.BPLCard))
443 {
444     using (var client = new WebClient())
445     {
446         string ClientID = SiteKeys.RationClientID;
447         //string ClientSecret = SiteKeys.RationClientSecret;
448         string rationCardNumber = model.CertificateNumber;
449         string RationAPIUrl = string.Format(SiteKeys.RationAPIUrl, rationCardNumber);
450         client.Headers.Add("Content-Type:application/json"); //Content-Type
451         client.Headers.Add("Accept:application/json");
452
453         client.Encoding = Encoding.UTF8;
454         var output = client.DownloadString(RationAPIUrl); //URI
455         SaveWrapperLog(model);
456
457         if (!string.IsNullOrWhiteSpace(output))
458         {
459             XmlDocument doc = new XmlDocument();
460             doc.LoadXml(output);
461             var output1 = JsonConvert.SerializeXmlNode(doc);
462             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output1, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
463         }
464         else
465         {
466             return new CustomActionResult<object>(HttpStatusCode.InternalServerError, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTION_OCCURRED });
467         }
468     }
469 }
470 else if (model.DocumentTypeID==Convert.ToInt32(DocumentTypeEnum.RBSC10thMarksheetMain))
471 {
472     DocumentVerificationEntities entities = new DocumentVerificationEntities();
473
474     if (model.RollNumber != "" && model.ExamType != 0 && model.ExamYear != "")
475     {
476         var getData = (from t1 in entities.tbl_OTVLocalVault

```

Document Verification API's Specification

3.13. Birth Certificate:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber Enter your Certificate Number

DocumentTypeId : 4

SSOID: Enter your SSOID

ProjectCode : Enter your Project code

IssueDate (Date): Enter your Issue date

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "Data": [  
        {  
            "REGISNUMBER": "0810700200000203364 / 2017",  
            "EventDate": "24/04/2017",  
            "Name": "DIVYAM JANGID",  
            "FatherName": "MANISH",  
            "MohterName": "REKHA",  
            "Address": "KAILASH NAGAR , हहण्‌टन / कर ली /  
राजस्थान", "Sex": "Male",  
            "Status": "1",  
            "Remarks": "Successfully Fetched"  
        }  
    ],  
    "DocumentName": "Birth Certificate",  
}
```

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Birth Certificate",  
    "Data": null  
}
```

Document Verification API's Specification

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the

Document Verification API's Specification

		data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



The screenshot shows a code editor with a C# file named `WrapperController.cs`. The code is part of a Web API project. It contains logic for handling document verification requests based on the document type. The code includes imports for `Convert.ToInt32`, `DocumentTypeEnum`, `HttpStatusCode`, `HttpResponseMessage`, and various service clients like `DateOfBirthDeathMarriageService` and `DataValidation.Core`. The logic involves checking if the issue date is null, creating a registration object, generating a salted password, and then searching for the registration. If successful, it saves the log and returns a response with success status and zero total records. If the document type is a death certificate, it returns a different response message.

```

643     }
644     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.BirthCertificate))
645     {
646         if (model.IssueDate == null)
647         {
648             return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Issue Date(mm/dd/yyyy)" });
649         }
650
651         DateOfBirthDeathMarriageService.searchregSoapClient o = new DateOfBirthDeathMarriageService.searchregSoapClient();
652         string RegistrationNumber, Year;
653         RegistrationNumber = model.CertificateNumber;
654
655         Year = DataValidation.Core.CoreExtensions.ConvertToDate(model.IssueDate).Year.ToString();
656         RegistrationNumber = model.CertificateNumber;
657
658         short Event = 1; //Birth
659         string RISLUserID = SiteKeys.RISLUserID;
660         string RISLPassword = SiteKeys.RISLPassword;
661         string SaltNo = "99999999";
662         string saltpassword = Common.GeneratePassword(RISLPassword, SaltNo);
663
664         var output = o.SearchRegistration(RISLUserID, saltpassword, SaltNo, Event, RegistrationNumber, Year);
665         if (output != null)
666         {
667             SaveWrapperLog(model);
668             var oRootObject = JsonConvert.DeserializeObject<List<BirthServiceModel>>(output);
669             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = oRootObject, IsSuccess = true, Message = CustomMessages.DATARETRIVED, TotalRecordsCount = 0 });
670         }
671
672         else
673         {
674             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
675         }
676     }
677     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.DeathCertificate))
678     {
679     }
680 }

```

Document Verification API's Specification

3.14. Death Certificate:

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeId : 5

SSOID: Enter your SSOID

ProjectCode : Enter your Project code

IssueDate: Enter your Issue date

Response json:

Successful Response:

```
{  
    "Message": "Data Retrieved",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "Data": [  
        {  
            "REGISNUMBER": "08110005430061500082 / 2018",  
            "EventDate": "29/09/2018",  
            "Name": "त०रा चाद",  
            "FatherName": "कानी दे वी",  
            "MohterName":  
                "हरालाताल",  
                "Address": "रातोंगरोता काठा मोहवात आजाताद चाक  
नरायनाता तहसील फलेरात जजलात जयपर  
303348",  
                "Sex": "Male",  
                "Status": "1",  
                "Remarks": "Successfully Fetched"  
        }  
    ],  
    "DocumentName": "Death Certificate",  
}
```

राजस्थान

Failure Response:

```
{  
    "Message": "Exception Occurred",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Death Certificate",  
    "Data": null  
}
```

Document Verification API's Specification

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type
IssueDate	int	This parameter indicates the certificate issue date and it will be a int type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string

Document Verification API's Specification

		type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:

```

    ...
    else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.DeathCertificate))
    {
        if (model.IssueDate == null)
        {
            return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Issue Date(mm/dd/yyyy)" });
        }
        DateOfBirthDeathMarriageService.searchregSoapClient o = new DateOfBirthDeathMarriageService.searchregSoapClient();
        string RegistrationNumber, Year;
        RegistrationNumber = model.CertificateNumber;
        Year = DateValidation.Core.CoreExtensions.ConvertToDate(model.IssueDate).Year.ToString();
        string Event = "/Death";
        string RISLUserID = SiteKeys.RISLUserID;
        string RISLPassword = SiteKeys.RISLPassword;
        string SaltNo = "99999999";
        string saltpassword = Common.GeneratePassword(RISLPassword, SaltNo);
        var output = o.SearchRegistration(RISLUserID, saltpassword, SaltNo, Event, RegistrationNumber, Year);
        if (output != null)
        {
            SaveWrapperLog(model);
            var oRootObject = JsonConvert.DeserializeObject<List<DeathServiceModel>>(output);
            return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = oRootObject, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
            //, TotalRecordsCount = totalRecordsCount
        }
        else
        {
            return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
        }
    }
    else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.MarriageCertificate))
    {
        if (model.MarriageDate == null)
        ...
    }
}

```

3.15. Marriage Certificate:

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeId : 2

SSOID: Enter your SSOID

ProjectCode : Enter your Project code

MarriageDate : Enter your Marriage date

Response json:

Successful Response:

Document Verification API's Specification

```
{  
    "Message": "Data Retrieved",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "Data": [  
        {  
            "REGISNUMBER": "08104007430048700009 / 2019",  
            "MarriageDate": "08/03/2019",  
            "GroomName": "गगन कमार शमा",  
            "GroomFatherName": "गगराज प्रसाद शमा",  
            "BrideName": "गृह्या शमा",  
            "BrideFatherName": "राजेत्र प्रसाद",  
            "MarriagePlaceAdd": "गाम झाकरा तहसील  
रायपाटी ज़िला अलवर ,", "Status": "1",  
            "Remarks": "Successfully Fetched"  
        }  
    ],  
    "DocumentName": "Marriage Certificate",  
}
```

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Marriage Certificate",  
    "Data": null  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Document Verification API's Specification

Marriage Date	int	This Parameter indicates the marriage date for this certificate and it will be a int type
---------------	-----	---

Successfully Response:

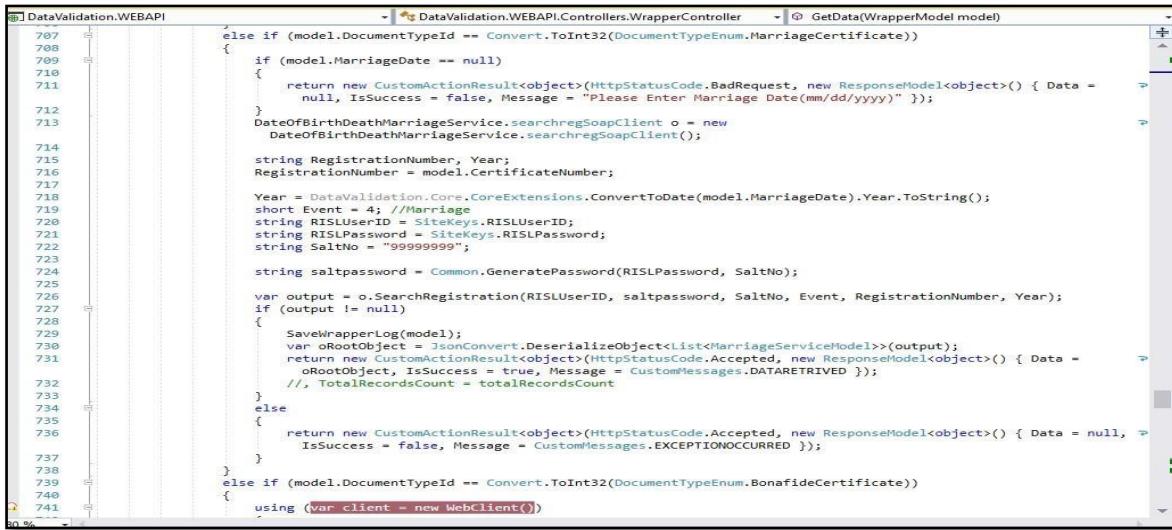
Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Code Snippet:



```

707     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.MarriageCertificate))
708     {
709         if (model.MarriageDate == null)
710         {
711             return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Marriage Date(mm/dd/yyyy)" });
712         }
713         DateOfBirthDeathMarriageService.searchregSoapClient o = new DateOfBirthDeathMarriageService.searchregSoapClient();
714         string RegistrationNumber, Year;
715         RegistrationNumber = model.CertificateNumber;
716         Year = DataValidation.Core.CoreExtensions.ConvertToDate(model.MarriageDate).Year.ToString();
717         short Event = 4; //Marriage
718         string RISLUserID = SiteKeys.RISLUserID;
719         string RISLPassword = SiteKeys.RISLPassword;
720         string SaltNo = "99999999";
721
722         string saltpassword = Common.GeneratePassword(RISLPassword, SaltNo);
723
724         var output = o.SearchRegistration(RISLUserID, saltpassword, SaltNo, Event, RegistrationNumber, Year);
725         if (output != null)
726         {
727             SaveWrapperLog(model);
728             var oRootObject = JsonConvert.DeserializeObject<List<MarriageServiceModel>>(output);
729             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = oRootObject, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
730             //, TotalRecordsCount = totalRecordsCount
731         }
732         else
733         {
734             return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
735         }
736     }
737     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.BonafideCertificate))
738     {
739         using (var client = new WebClient())
740     }

```

3.16. Bonafide Certificate:

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number
DocumentTypeId : 9
SSOID: Enter your SSOID
ProjectCode : Enter your Project Code

Response json:

Successful Response:

```
{
    "Message": "Data Retrived",
    "IsSuccess": true,
    "TotalRecordsCount": 1,
    "DocumentName": "Bonafide Certificate",
    "Data": {
        "Status": "Approved",
        "Certificate_url":
        "https://emitraapp.rajasthan.gov.in/emitrashared/uploadeddocs/tempCertEsignedN/05-02-2019/cnSoQq+7trOrxNLPfPGSEA==/180092709362_Cert.pdf",
        "Current_Address": "Ramdhan, Ward 5, Sureli, Uniara, Tonk",
        "Digitally_Signed": "Yes",
        "Permanent_Address": "Ramdhan, Ward 5, Sureli, Uniara, Tonk",
        "Date_Of_Birth": "01-Jun-1995",
        "Approved_And_Signed_By": "Office of Tehsildar, Uniara, Tonk",
        "Applicant_Name": "Mayaram Meena",
        "Father_Name": "Ramdhan Meena",
        "Mother_Name": "Kadi Devi",
    }
}
```

Document Verification API's Specification

```

    "TOKEN_NO": "180092709362",
    "Approved_Date": "2019-02-05 10:39:50.0",
    "ISAVAILABLE": "Y"
}
}

```

Failure Response:

```

{
    "Message": "Data Not Found",
    "IsSuccess": false,
    "TotalRecordsCount": 0,
    "DocumentName": "Bonafied Certificate",
    "Data": null
}

```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type

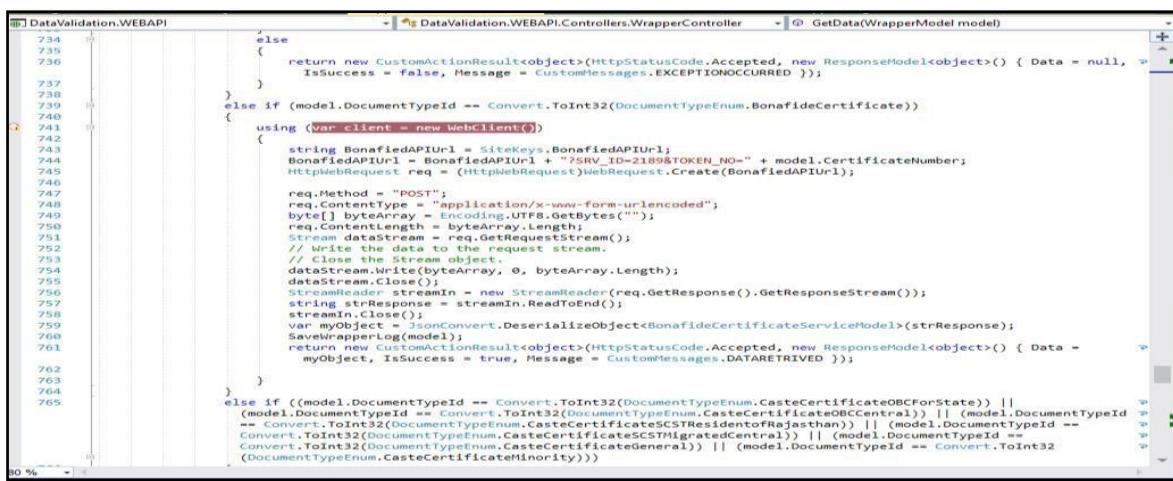
Document Verification API's Specification

DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



```

    734
    735
    736
    737
    738
    739
    740
    741
    742
    743
    744
    745
    746
    747
    748
    749
    750
    751
    752
    753
    754
    755
    756
    757
    758
    759
    760
    761
    762
    763
    764
    765
  
```

```

    }
    else
    {
        return new CustomActionResult<object>(HttpStatusCodes.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTION_OCCURRED });
    }
}
else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.BonafideCertificate))
{
    using (var client = new WebClient())
    {
        string BonafideAPIUrl = SiteKeys.BonafideAPIUrl;
        BonafideAPIUrl = BonafideAPIUrl + "5SRV_ID=21898TOKEN_NO=" + model.CertificateNumber;
        HttpWebRequest req = (HttpWebRequest)WebRequest.Create(BonafideAPIUrl);

        req.Method = "POST";
        req.ContentType = "application/x-www-form-urlencoded";
        byte[] byteArray = Encoding.UTF8.GetBytes("");
        req.ContentLength = byteArray.Length;
        Stream dataStream = req.GetRequestStream();
        // Write the data to the request stream.
        dataStream.Write(byteArray, 0, byteArray.Length);
        dataStream.Close();
        Stream streamIn = new StreamReader(req.GetResponse().GetResponseStream());
        string strResponse = streamIn.ReadToEnd();
        streamIn.Close();
        var myObject = JsonConvert.DeserializeObject<BonafideCertificateServiceModel>(strResponse);
        SaveWrapperLog(model);
        return new CustomActionResult<object>(HttpStatusCodes.Accepted, new ResponseModel<object>() { Data = myObject, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
    }
}
else if ((model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.CasteCertificateOBCForState)) ||
(model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.CasteCertificateOBCCentral)) || (model.DocumentTypeId ==
Convert.ToInt32(DocumentTypeEnum.CasteCertificateSCSTResidentofRajasthan)) || (model.DocumentTypeId ==
Convert.ToInt32(DocumentTypeEnum.CasteCertificateSCSTMigratedCentral)) || (model.DocumentTypeId ==
Convert.ToInt32(DocumentTypeEnum.CasteCertificateGeneral)) || (model.DocumentTypeId == Convert.ToInt32(
(DocumentTypeEnum.CasteCertificateMinority)))
)
{
    using (var client = new WebClient())
    {
        string BonafideAPIUrl = SiteKeys.BonafideAPIUrl;
        BonafideAPIUrl = BonafideAPIUrl + "5SRV_ID=21898TOKEN_NO=" + model.CertificateNumber;
        HttpWebRequest req = (HttpWebRequest)WebRequest.Create(BonafideAPIUrl);

        req.Method = "POST";
        req.ContentType = "application/x-www-form-urlencoded";
        byte[] byteArray = Encoding.UTF8.GetBytes("");
        req.ContentLength = byteArray.Length;
        Stream dataStream = req.GetRequestStream();
        // Write the data to the request stream.
        dataStream.Write(byteArray, 0, byteArray.Length);
        dataStream.Close();
        Stream streamIn = new StreamReader(req.GetResponse().GetResponseStream());
        string strResponse = streamIn.ReadToEnd();
        streamIn.Close();
        var myObject = JsonConvert.DeserializeObject<BonafideCertificateServiceModel>(strResponse);
        SaveWrapperLog(model);
        return new CustomActionResult<object>(HttpStatusCodes.Accepted, new ResponseModel<object>() { Data = myObject, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
    }
}

```

Document Verification API's Specification

3.17. Labour Card:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber: Enter your Certificate Number

DocumentTypeId : 21

SSOID: Enter your SSOID

ProjectCode: Enter your Project Code

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "Data": [  
        {  
            "act": "Beneficiary Registration",  
            "reg_num": "B1/2016/0018814",  
            "reg_date": "2/10/2016 9:30:51 PM",  
            "name": "Yugul Kishore",  
            "status": "Accepted",  
            "AppStatus": "1",  
            "ExpiryDate": "2/23/2021 12:00:00 AM",  
            "ActID": "0",  
            "CertificatePath": ""  
        }  
    ],  
    "DocumentName": "Labour Card",  
}
```

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Labour Card",  
    "Data": null  
}
```

Document Verification API's Specification

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:



```

321     jsonString += "}";
322     string strResponse = jsonString;
323     SaveWrapperLog(model);
324     return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = strResponse, IsSuccess =
325         = true, Message = CustomMessages.DATARETRIVED });
326 }
327 else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.labourCard))
328 {
329     LabourCardServiceReference.CheckRegistrationSoapClient obj = new LabourCardServiceReference.CheckRegistrationSoapClient();
330     string RegistrationNumber;
331     RegistrationNumber = model.CertificateNumber;           // "B1/2016/0018814"
332     string UserId = SiteKeys.labourCardUserName;
333     string Password = SiteKeys.labourCardPassword;
334
335     var output = obj.CheckRegistrationStatusJSON(UserId, Password, RegistrationNumber);
336     if (output != null)
337     {
338         SaveWrapperLog(model);
339         var oRootObject = JsonConvert.DeserializeObject<List<LabourCardServiceModel>>(output);
340         return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = oRootObject,
341             IsSuccess = true, Message = CustomMessages.DATARETRIVED });
342     }
343     else
344     {
345         return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess =
346             false, Message = CustomMessages.EXCEPTIONOCCURRED });
347     }
348 }
349 else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.PoliceCharacterCertificate))
350 {
351 }

```

Document Verification API's Specification

3.18. Caste Certificate:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber : Enter your Certificate Number

DocumentTypeId : 23, 24, 25, 26, 27, 28

SSOID: Enter your SSOID

ProjectCode : Enter your Project Code

Note: - (Document type Id pass only one at a time.)

Document Type ID	Document Type ID Description
23	This document type id will be used for the OBC caste certificate (State).
24	This document type id will be used for the OBC caste certificate (Central).
25	This document type id will be used for SC, ST caste certificate (Resident of Rajasthan).
26	This document type id will be used for SC, ST caste certificate (Migrated Central).
27	This document type id will be used for the General caste certificate.
28	This document type id will be used for the Minority caste certificates.

Response json:

Successful Response:

```
{  
    "Message": "Data Retrived",  
    "IsSuccess": true,  
    "TotalRecordsCount": 1,  
    "DocumentName": "Caste Certificate",  
    "Data": {  
        "Status": "Approved",  
        "Certificate_url":  
            "https://emitraapp.rajasthan.gov.in/emitrashared/uploadeddocs/tempCertDigital/18-10-  
            2019/mK4IJ4eHs6F1eyrirOvnpQ==/190239813950_Cert.pdf",  
        "Approved_And_Signed_By": "Office of the Sub Divisional  
Officer,Nawalgarh,(Jhunjhunu)",  
        "Applicant_Name": "naveen ranwa",  
        "Current_Address": "vpo-nawalri",  
        "Digitally_Signed": "Yes",  
        "Permanent_Address": "vpo-nawalri",  
        "TOKEN_NO": "190239813950",  
        "Date_Of_Birth": "20-Oct-2002",  
        "Approved_Date": "2019-10-18 15:55:28.0",  
        "ISAVAILABLE": "Y",  
        "CasteList": 0  
    }  
}
```

Document Verification API's Specification

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Caste Certificate",  
    "Data": null  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an

Document Verification API's Specification

	string type
--	-------------

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Code Snippet:

```

    ...
    else if ((model.DocumentTypeId == Convert.ToInt32(DocumentTypeInnum.CasteCertifiedForState)) || (model.DocumentTypeId == Convert.ToInt32(DocumentTypeInnum.CasteCertifiedForCentral)) || (model.DocumentTypeId == Convert.ToInt32(DocumentTypeInnum.CasteCertifiedSCSThroughoutIndia)) || (model.DocumentTypeId == Convert.ToInt32(DocumentTypeInnum.CasteCertifiedGeneral)))
    {
        serviceId = "J2317";
    }
    if (dec1d == 22)
    {
        serviceId = "J2317";
    }
    if (dec1d == 23)
    {
        serviceId = "J2317";
    }
    if (dec1d == 24)
    {
        serviceId = "J2317";
    }
    if (dec1d == 25)
    {
        serviceId = "J2317";
    }
    if (dec1d == 26)
    {
        serviceId = "J2317";
    }
    if (dec1d == 27)
    {
        serviceId = "J2317";
    }
    if (dec1d == 28)
    {
        serviceId = "J2317";
    }
    if (dec1d == 29)
    {
        serviceId = "J2317";
    }

    string NonPaidFieldID = stringKeys.BenefidAPIDURL;
    NonPaidFieldID += NonPaidFieldID + "J2317_ID" + serviceId + "B00000_No" + model.CertificateNumber;
    string NonPaidFieldID = NonPaidFieldID + "J2317_ID" + serviceId + "B00000_No" + model.CertificateNumber;

    req.Method = "POST";
    req.ContentType = "application/x-www-form-urlencoded";
    byte[] bytes = Encoding.UTF8.GetBytes("");
    req.ContentLength = bytes.Length;
    req.InputStream = new MemoryStream(bytes);
    // write the data to the request stream.
    // Close the Stream object.
    stream.Close();
    datastream = new FileStream("D:\\" + NonPaidFieldID + ".pdf", FileMode.Create);
    datastream.Close();
    // Create a new byte array for reading the response.
    byte[] buffer = new byte[1024];
    int bytesRead;
    while ((bytesRead = stream.Read(buffer, 0, 1024)) > 0)
    {
        datastream.Write(buffer, 0, bytesRead);
    }
    datastream.Close();
    stream.Close();
    string strResponse = new StreamReader(req.GetResponse().GetResponseStream());
    string strResponse = strResponse.ReadToEnd();
    if (!string.IsNullOrEmpty(strResponse))
    {
        // Saving response in file
        File.WriteAllText("D:\\" + NonPaidFieldID + ".pdf", strResponse);
        // Deserializing response
        var objDeserializedObject = JsonConvert.DeserializeObject<CustomActionResultObject>(strResponse);
        var objAccepted = JsonConvert.DeserializeObject<CustomActionResultObject>(objDeserializedObject.Accepted);
        var objRejected = JsonConvert.DeserializeObject<CustomActionResultObject>(objDeserializedObject.Rejected);
        if (objAccepted != null)
        {
            return new CustomActionResultObject<CustomActionResultObject>(HttpStatusCode.Accepted, new ResponseModelObject<CustomActionResultObject>() { Data = objAccepted, IsSuccess = true, Message = CustomMessages.DATARETRIVED });
        }
        else
        {
            return new CustomActionResultObject<CustomActionResultObject>(HttpStatusCode.InternalServerError, new ResponseModelObject<CustomActionResultObject>() { Data = null, IsSuccess = false, Message = CustomMessages.DATANOTFOUND });
        }
    }
    return new CustomActionResultObject<CustomActionResultObject>(HttpStatusCode.InternalServerError, new ResponseModelObject<CustomActionResultObject>() { Data = null, IsSuccess = false, Message = CustomMessages.DATANOTFOUND });
}

```

Activate Windows

3.19. Registration Certificate (Vehicle):

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber: Enter your Certificate Number

DocumentTypeId : 17

SSOID: Enter your SSOID

Document Verification API's Specification

ProjectCode : Enter your Project Code

Response json:

Successful Response:

```
{
  "Message": "Data Retrived",
  "IsSuccess": true,
  "TotalRecordsCount": 1,
  "DocumentName": "Registration Certificate",
  "Data": "{\"?xml\":{\"@version\":\"1.0\",\"@encoding\":\"UTF-8\"},\"@standalone\":\"yes\"},\"VehicleDetails\":{\"stautsMessage\":\"OK\",\"rc_regn_no\":\"RJ02BG3417\",\"rc_owner_sr\":1,\"rc_owner_name\":\"GAGAN KUMAR SHARMA\",\"rc_present_address\":\"VILL-PATAN, ALWAR, , Alwar-301414\",\"rc_vh_class_desc\":\"M-Cycle/Scooter(2WN)\",\"rc_chasi_no\":\"ME4JC65AMJ7270347\",\"rc_eng_no\":\"JC65E72423390\",\"rc_maker_desc\":\"HONDA MOTORCYCLE AND SCOOTER INDIA (P) LTD\",\"rc_maker_model\":\"CB SHINE\",\"rc_fit_upto\":\"12-Feb-2034\",\"rc_financer\":\"\",\"rc_insurance_comp\":\"\",\"rc_insurance_policy_no\":\"\",\"rc_insurance_upto\":\"\",\"rc_registered_at\":\"ALWAR RTO, Rajasthan\",\"rc_status_as_on\":\"06-Aug-2021\",\"rc_status\":\"ACTIVE\"}}"
}
```

Failure Response:

```
{
  "Message": "Data Retrived",
  "IsSuccess": true,
  "TotalRecordsCount": 0,
  "DocumentName": "Registration Certificate",
  "Data": "{\"?xml\":{\"@version\":\"1.0\",\"@encoding\":\"UTF-8\"},\"@standalone\":\"yes\"},\"VehicleDetails\":{\"stautsMessage\":\"Vehicle Data Not Found\"}"
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type

Document Verification API's Specification

ProjectCode	String	This parameter indicates the Project code and It will be a string type
-------------	--------	--

Successfully Response:

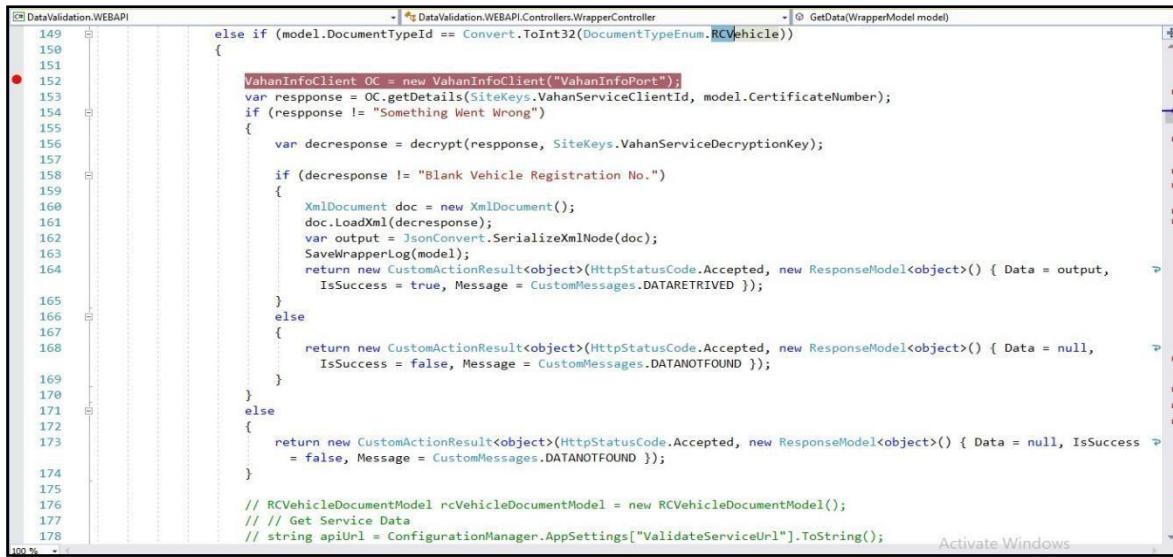
Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Code Snippet:



The screenshot shows a code editor window with the following details:

- Title Bar:** DataValidation.WEBAPI - DataValidation.WEBAPI.ControllersWrapperController - GetData(WrapperModel model)
- Code Area:** C# code for a controller method named `GetData`. The code handles document verification for different types, specifically focusing on vehicle documents.
- Code Snippet:**

```
else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.RCVehicle))
{
    VahanInfoClient OC = new VahanInfoClient("VahanInfoPort");
    var response = OC.getDetails(SiteKeys.VahanServiceClientId, model.CertificateNumber);
    if (response != "Something Went Wrong")
    {
        var decresponse = decrypt(response, SiteKeys.VahanServiceDecryptionKey);

        if (decresponse != "Blank Vehicle Registration No.")
        {
            XmlDocument doc = new XmlDocument();
            doc.LoadXml(decresponse);
            var output = JsonConvert.SerializeXmlNode(doc);
            SaveWrapperLog(model);
            return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = output,
                IsSuccess = true, Message = CustomMessages.DATARETRIVED });
        }
        else
        {
            return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null,
                IsSuccess = false, Message = CustomMessages.DATANOTFOUND });
        }
    }
    else
    {
        return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess =
            = false, Message = CustomMessages.DATANOTFOUND });
    }

    // RCVehicleDocumentModel rcVehicleDocumentModel = new RCVehicleDocumentModel();
    // // Get Service Data
    // string apiUrl = ConfigurationManager.AppSettings["ValidateServiceUrl"].ToString();
}
```
- Status Bar:** Activate Windows

3.20. Disability Certificate:

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

CertificateNumber: Enter your Certificate Number

DocumentTypeId : 20,

SSOID: Enter your SSOID

ProjectCode : Enter your Project Code

Response json:

Successful Response:

```
{
    "Message": "Data Retrived",
    "IsSuccess": true,
    "TotalRecordsCount": 1,
    "DocumentName": "Disability Certificate",
    "Data": "\r\n \"Valid\": true,\r\n \"Description\": \"Data Send successfully.\",\r\n \"Value\": [{\"CERTIFICATEPATH\": \"https://dsa.rajasthan.gov.in/EnglishCertificate/Form-IV/SAP_2017_152.pdf\", \"APPLICANTNAME\": \"SOURABH\", \"APPLICANTNAMEHindi\": \"\u091c\u093f\u093e\u0930\u093f\u093e\u0930\u093f\u093e\u0930\", \"FATHERNAME\": \"PRITHVI RAJ\", \"FATHERNAMEHindi\": \"\u091c\u093f\u093e\u0930\u093f\u093e\u0930\u093f\u093e\u0930\", \"DATEOFBIRTH\": \"06-08-1993 00:00:00\", \"UDIDNo\": \"SAP/2017/152\", \"STREETROAD\": \"WARD
```

Document Verification API's Specification

```
NO.11\" , "LANDMARK": "SANGARIA", "AREALOCALITY": "SANGARIA", "StateName": "RAJASTHAN", "DistrictNameENg": "HANUMANGARH", "DistrictNameHindi": "ହନୁମାର୍ଗ", "BlockNameEng": "RAJGARH", "BlockNameHindi": "ରାଜଗର୍ହ", "CityNameEng": "SANGARIA", "CityNameHindi": "ଶାଂତିବାସ", "GPNameEng": "CHEETIWAS", "GPNameHindi": "ଚେତିବାସ", "WardNameEng": "WARD NO- 11", "WardNameHindi": "ପ୍ଲଟ୍ ନଂ - 11", "VillageNameEng": "", "VillageNameHindi": "", "PINCODE": "335063", "RURALURBAN": "URBAN", "AasthaCardIssued": "0", "DisabilityType": "LOCOMOTOR DISABILITY", "DisabilityPercentage": "50"}]\r\n}
```

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 0,  
    "DocumentName": "Disability Certificate",  
    "Data": null  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
CertificateNumber	String	This parameter indicates the Certificate Number and It will be a string type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Document Verification API's Specification

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Code Snippet:

```

348         }
349     }
350     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.RBSC12thMarksheetMain))
351     {
352     }
353     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.DisabilityCertificate))
354     {
355         using (var client = new WebClient())
356         {
357             DisabilityCertificateServiceModel disabilityCertificateServiceModel = new DisabilityCertificateServiceModel();
358             string SapNumber = model.CertificateNumber;
359             string AadhaarNumber = model.CertificateNumber;
360             string DisabilityCertificateAPIURL = string.Format(SiteKeys.DisabilityCertificateAPIURL, SapNumber);
361             DisabilityCertificateAPIURL = DisabilityCertificateAPIURL + "Aadhar=" + AadhaarNumber;
362             client.Headers.Add("Content-Type:application/json"); //Content-Type
363             client.Encoding = Encoding.UTF8;
364             var output = client.DownloadString(DisabilityCertificateAPIURL); //URI
365             string outputobject = JsonConvert.SerializeObject(output);
366             SaveWrapperLog(model);
367             if (!string.IsNullOrWhiteSpace(outputobject))
368             {
369                 //var oRootObject = JsonConvert.DeserializeObject<PoliceVerificationServiceModel>(output);
370                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = outputobject, IsSuccess = true, Message = CustomMessages.DATARETRIEVED });
371             }
372             else
373             {
374                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = CustomMessages.EXCEPTIONOCCURRED });
375             }
376         }
377     }
378     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.BPLCard))
379     {
380         using (var client = new WebClient())
381         {
382             string ClientID = Sitekeys.RationClientID;
383             //string ShmseshalNo = "0000-111-0005";
384             string rationCardNumber = model.CertificateNumber;
385             string RationAPIUrl = string.Format(SiteKeys.RationAPIUrl, rationCardNumber);
386             client.Headers.Add("Content-Type:application/json"); //Content-Type
387         }
388     }
389 }
390 }
391 }
392 }
393 }
394 }
395 }
396 }
397 }
398 }
399 }
400 }
401 }
402 }
403 }
404 }
405 }
406 }
407 }
408 }
409 }
410 }
411 }
412 }
413 }
414 }
415 }
416 }
417 }
418 }
419 }
420 }
421 }
422 }
423 }
424 }
425 }
426 }
427 }
428 }
429 }
430 }
431 }
432 }
433 }
434 }
435 }
436 }
437 }
438 }
439 }
440 }
441 }
442 }
443 }
444 }
445 }
446 }
447 }
448 }
449 }
450 }
451 }
452 }
453 }

```

3.21. CBSC Marksheets:

Private URL: <http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

DocumentTypeId:41

SSOID: Enter yourSSOID

ProjectCode : Enter your Project Code

FullName: Enter your Full Name

RollNumber:Enter your Roll Number

ExamType:Enter your Exam Type

ExamYear:Enter your Exam Year

Response json:

Successful Response:

```
{
    "Message": "Data Retrieved",
    "IsSuccess": true,
    "TotalRecordsCount": 1,
    "DocumentName": "CBSE Marksheets",
    "Data": {

```

Document Verification API's Specification

```
"docXML": "<?xml version=\"1.0\" encoding=\"utf-8\"?>\n<Certificate language=\"99\"\nname=\"Class X Marksheets\" type=\"SSCER\" number=\"2130654\" issuedAt=\"\"\nissueDate=\"\" validFromDate=\"\" status=\"A\"><IssuedBy><Organization name=\"Central\nBoard of Secondary Education\" code=\"\" tin=\"\" uid=\"\" type=\"CG\"><Address\ntype=\"\" line1=\"\" line2=\"\" house=\"\" landmark=\"\" locality=\"\" vtc=\"\" district=\"\"\npin=\"\" state=\"\" country=\"IN\"/></Organization></IssuedBy><IssuedTo><Person\nuid=\"\" title=\"\" name=\"EKNOOR SINGH\" dob=\"08-12-2001\" swd=\"RUPINDER\nSINGH\" swdIndicator=\"\" motherName=\"AMIT PAL KAUR\" gender=\"Male\"\nmaritalStatus=\"\" religion=\"\" phone=\"\" email=\"\"><Address type=\"permanent\"\nline1=\"\" line2=\"\" house=\"\" landmark=\"\" locality=\"\" vtc=\"\" district=\"\" pin=\"\"\nstate=\"\" country=\"IN\"/><Photo\nformat=\"\"/></Person></IssuedTo><CertificateData><School name=\"DELHI PUBLIC\nSCHOOL SECTOR 45 U.E. GURGAON HRY\" code=\"20020\" regionName=\"\"\nregionCode=\"C\"/><Examination name=\"SECONDARY SCHOOL EXAMINATION\"\ncenterCode=\"\" admitCardId=\"\" month=\"March\" year=\"2018\"/><Performance\nresult=\"PASS\" marksTotal=\"496\" marksMax=\"\" percentage=\"\" cgpa=\"\"\ncgpaMax=\"\" resultDate=\"\" updateDate=\"01-01-1970\"><Subjects><Subject\nname=\"ENGLISH COMM.\" code=\"101\" marksTheory=\"079\" marksMaxTheory=\"\"\nmarksPractical=\"020\" marksMaxPractical=\"\" marksTotal=\"099\" marksMax=\"\"\ngp=\"\" gpMax=\"\" grade=\"A1\"/><Subject name=\"HINDI COURSE-B\" code=\"085\"\nmarksTheory=\"078\" marksMaxTheory=\"\" marksPractical=\"020\"\nmarksMaxPractical=\"\" marksTotal=\"098\" marksMax=\"\" gp=\"\" gpMax=\"\"\ngrade=\"A1\"/><Subject name=\"MATHEMATICS\" code=\"041\" marksTheory=\"080\"\nmarksMaxTheory=\"\" marksPractical=\"020\" marksMaxPractical=\"\" marksTotal=\"100\"\nmarksMax=\"\" gp=\"\" gpMax=\"\" grade=\"A1\"/><Subject name=\"SCIENCE\"\ncode=\"086\" marksTheory=\"079\" marksMaxTheory=\"\" marksPractical=\"020\"\nmarksMaxPractical=\"\" marksTotal=\"099\" marksMax=\"\" gp=\"\" gpMax=\"\"\ngrade=\"A1\"/><Subject name=\"SOCIAL SCIENCE\" code=\"087\" marksTheory=\"080\"\nmarksMaxTheory=\"\" marksPractical=\"020\" marksMaxPractical=\"\" marksTotal=\"100\"\nmarksMax=\"\" gp=\"\" gpMax=\"\" grade=\"A1\"/><Subject name=\"\" code=\"\"\nmarksTheory=\"\" marksMaxTheory=\"\" marksPractical=\"\" marksMaxPractical=\"\"\nmarksTotal=\"\" marksMax=\"\" gp=\"\" gpMax=\"\" grade=\"\"/><Subject name=\"INFO.\n& COMM. TECH\" code=\"166\" marksTheory=\"040\" marksMaxTheory=\"\"\nmarksPractical=\"060\" marksMaxPractical=\"\" marksTotal=\"100\" marksMax=\"\"\ngp=\"\" gpMax=\"\"\ngrade=\"A1\"/></Subjects></Performance></CertificateData></Certificate>\n",\n    "docPDF": \"PDF\",\n    "state": \"Wrapper API\",\n    "rollno": \"2130654\",\n    "status": \"Y\"\n}\n}
```

Document Verification API's Specification

Failure Response:

```
{  
    "Message": "Data Not Found",  
    "IsSuccess": false,  
    "TotalRecordsCount": 1,  
    "DocumentName": "CBSE Marksheets",  
    "Data": {  
        "docXML": "{\"error\":\"invalid_parameter\",\"errorDescription\":\"Bad request- Name  
Mismatched! Try Again using correct data...\"}\n",  
        "docPDF":  
        "eyJlcnjvcil6lmludmFsaWRfcGFyYW1ldGVyliwiZXJyb3JEZXNjcmIwdGlvbil6lkjhZCByZXF1ZXNO  
LSBOYW1lIE1pc21hdGNoZWQhIFRyeSBBZ2FpbIB1c2luZyBjb3JyZWN0IGRhGEuLi4ifQo=",  
        "state": "Wrapper API",  
        "rollno": "2130654",  
        "status": "Y"  
    }  
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
Full Name	String	This parameter indicates the full name and It will be a string type
Roll Number	int	This parameter indicates the roll number and It will be a int type
Exam Type	int	This parameter indicates the exam type and It will be a int type
Exam Year	int	This parameter indicates the exam year and It will be a int type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type

Document Verification API's Specification

ProjectCode	String	This parameter indicates the Project code and It will be a string type
-------------	--------	--

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Code Snippet:

```

    878     else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.CBSEMarksheet))
    879     {
    880         using (var client = new WebClient())
    881         {
    882             if (model.Rollnumber == null)
    883             {
    884                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Roll Number" });
    885             }
    886             if (model.ExamType == 0)
    887             {
    888                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Exam Type" });
    889             }
    890             if (model.ExamYear == null)
    891             {
    892                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Exam Year" });
    893             }
    894             if (model.FullName == null)
    895             {
    896                 return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = null, IsSuccess = false, Message = "Please Enter Full Name" });
    897             }
    898             cbsemk.modeldata = new cbsemkmodel();
    899             modeldata.docType = "both";
    900             modeldata.educlass = model.Examtype;
    901             modeldata.examyear = model.ExamYear;
    902             modeldata.rollno = model.Rollnumber;
    903             modeldata.FullName = model.FullName;
    904             modeldata.state = "Wrapper API";
    905             modeldata.applname = "Document Verification";
    906             string inputJson = JsonConvert.SerializeObject(modeldata);
    907             var myDeserializedClass = JsonConvert.DeserializeObject<RootObject>(output);
    908             if (myDeserializedClass.docXML.Contains("Error"))
    909             {
    910                 model.TransactionStatus = Convert.ToInt32(APITransactionStatus.DATANOTFOUND);
    911                 SaveWrapperLog(model);
    912             }
    913         }
    914     }
    915     var output = client.UploadString(CBSECertificateAPIUrl, "Post", inputJson);
    916     var myDeserializedClass = JsonConvert.DeserializeObject<RootObject>(output);
    917     if (myDeserializedClass.docXML.Contains("Error"))
    918     {
    919         model.TransactionStatus = Convert.ToInt32(APITransactionStatus.DATANOTFOUND);
    920         SaveWrapperLog(model);
    921     }
    922 }

```

3.22. JamaBandi Khata:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

DocumentTypeId : 39

SSOID: Enter your SSOID

ProjectCode : Enter your Project Code

districtId: Enter your District Id

tehsillId: Enter your Tehsil ID

VillId: Enter your Vill ID

UserName: Enter your User Name

Password: Enter your Password

Response json:

Successful Response:

```
{
    "Message": "Data Retrived",
    "IsSuccess": true,
    "TotalRecordsCount": 1,
    "DocumentName": "Jamabandi Khata List",
    "Data": [
        {
            "confirm": 1,
            "msg": "",
            "Khata": 1
        }
    ]
}
```

Failure Response:

{

Document Verification API's Specification

```
"Message": "Data Not Found",
"IsSuccess": false,
"TotalRecordsCount": 0,
"DocumentName": "Jamabandi Khata List",
"Data": []
}
```

Description:

Request Parameters:

Parameter Name	Data Type	Description
UserName	String	This parameter indicates the User Name and It will be a string type
Password	String	This parameter indicates the Password and It will be a string type
DistrictId	int	This parameter indicates the District id and it will be a int type
TehsilId	int	This parameter indicates the tehsil id and it will be a int type
VillId	int	This parameter indicates the village id and it will be a int type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Document Verification API's Specification

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.
Data	String	This parameter indicates the null data and it will be an string type

Document Verification API's Specification

Code Snippet:

```

907         }
908         else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeEnum.Jamabandikhata))
909     {
910         if (model.districtId == null)
911     {
912         return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, IsSuccess = false, Message = "Pls enter District Id (districtId)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
913     }
914     if (model.tehsillId == null)
915     {
916         return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, IsSuccess = false, Message = "Pls enter Tehsil Id (tehsillId)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
917     }
918     if (model.VillId == null)
919     {
920         return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, IsSuccess = false, Message = "Pls enter Village Id (VillId)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
921     }
922     if (model.UserName == null)
923     {
924         return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, IsSuccess = false, Message = "Pls enter User Name (UserName)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
925     }
926     if (model.Password == null)
927     {
928         return new CustomActionResult<object>(HttpStatusCode.BadRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, IsSuccess = false, Message = "Pls enter Password", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
929     }
930     ServicePointManager.ServerCertificateValidationCallback = new System.Net.Security.RemoteCertificateValidationCallback(delegate { return true; });
931     JamabandiService.ws_rajsewadwarSoapClient j = new JamabandiService.ws_rajsewadwarSoapClient();
932     var Khatalist=j.GetKhatalist(model.districtId, model.tehsillId, model.VillId, model.UserName, model.Password);
933     SaveWrapperLog(model);
934     return new CustomActionResult<object>(HttpStatusCode.Accepted, new ResponseModel<object>() { Data = Khatalist, TotalRecordsCount = Khatalist.Rows.Count, IsSuccess = true, Message = CustomMessages.DATARETRIVED, DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });

```

3.23. Jamabandi ROR:

Private URL:

<http://10.70.241.253/DOCUMENTVERIFICATIONENGINEWEBAPI/Wrapper/GetData>

Request Parameters:

DocumentTypeId : 40

SSOID: Enter your SSOID

ProjectCode: Enter your Project Code

districtId: Enter your District ID

tehsillId: Enter your Tehsil ID

VillId: Enter your Vill ID

UserName: Enter your User Name

Password: Enter your Password

ApplicantName: Enter your Application Name

ApplicantMobile: Enter your Application Mobile

ApplicantAddress: Enter your Applicant Address

Response json:

Successful Response:

```
{
    "Message": "Data Retrived",
    "IsSuccess": true,
    "TotalRecordsCount": 1,
    "DocumentName": "Jamabandi ROR",
    "Data": [
        {
            "khata": "5",
            "rorbase64": "

```

Document Verification API's Specification

```
"JVBERi0xLjMNCjEgMCBvYmoNClsvUERGIC9UZXh0IC9JbWFnZUIgLOltYWdlQyAvSW1hZ2VJX
Q0KZW5kb2JqDQo5IDAcb2JqDQo8PCAvTGVuZ3RoIDMwMjQgLOZpbHRlcAvRmxhdGVEZWN
vZGUgPj4gc3RyZWftDQpYCcd1cW6sjuRF+H5j/0I...
}]
}
```

Failure Response:

```
{
  "Message": "Data Not Found",
  "IsSuccess": false,
  "TotalRecordsCount": 0,
  "DocumentName": "Jamabandi ROR",
  "Data": []
}
```

Description:**Request Parameters:**

Parameter Name	Data Type	Description
UserName	String	This parameter indicates the User Name and It will be a string type
Password	String	This parameter indicates the Password and It will be a string type
ApplicantName	String	This parameter indicates the Applicant name and It will be a string type
ApplicantMobile	int	This parameter indicates the Applicant Mobile number and it will be a int type
ApplicantAddress	String	This parameter indicates the Applicant address and It will be a string type
Khata	int	This parameter indicates the Khata and it will be a int type
DistrictId	int	This parameter indicates the District id and it will be a int type
TehsilId	int	This parameter indicates the tehsil id and it will be a int type

Document Verification API's Specification

VillId	int	This parameter indicates the village id and it will be a int type
DocumentTypeId	int	This parameter indicates the Document type id and it will be a int type
SSOID	String	This parameter indicates the SSOID and It will be a string type
ProjectCode	String	This parameter indicates the Project code and It will be a string type

Successfully Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the message of data retrieved and it will be a string type
IsSuccess	String	This parameter checks the data and if it is successful response than it will be true
TotalRecordsCount	String	This parameter indicates the total record count and it will be a string type
DocumentName	String	This parameter shows the document name and it will be a string type
Data	String	This parameter indicates the null data and it will be an string type

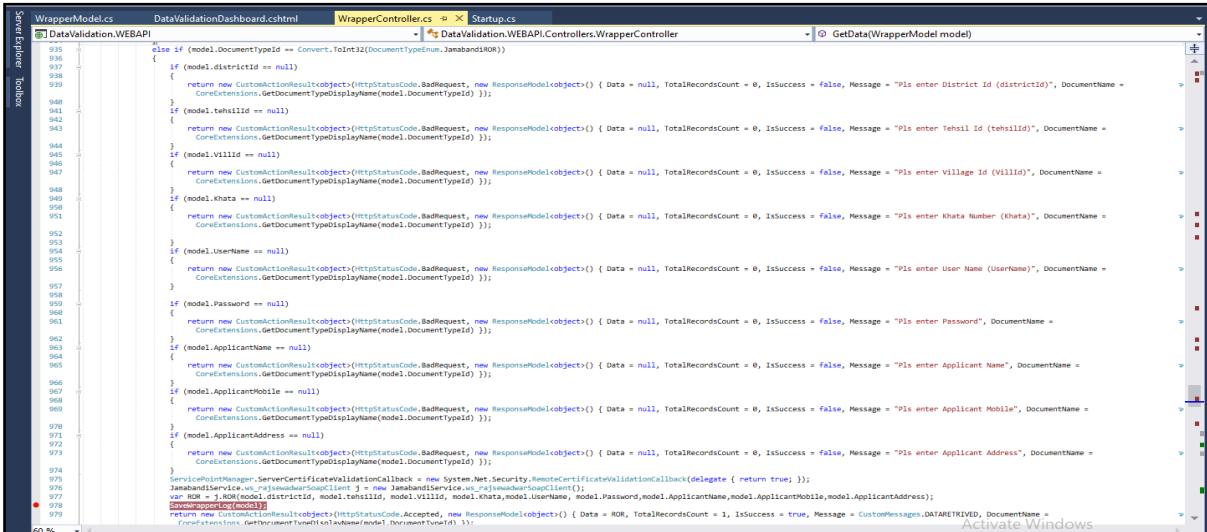
Failure Response:

Parameter Name	Data Type	Description
Message	String	This parameter indicates the error and it will be a string type
IsSuccess	String	This parameter checks the data and if it is unsuccessful response than it will be false
TotalRecordsCount	String	This parameter indicates the total records count, if total records count will be wrong.
DocumentName	String	This parameter indicates the document name and it will be an string type.

Document Verification API's Specification

Data	String	This parameter indicates the null data and it will be an string type
-------------	---------------	--

Code Snippet:

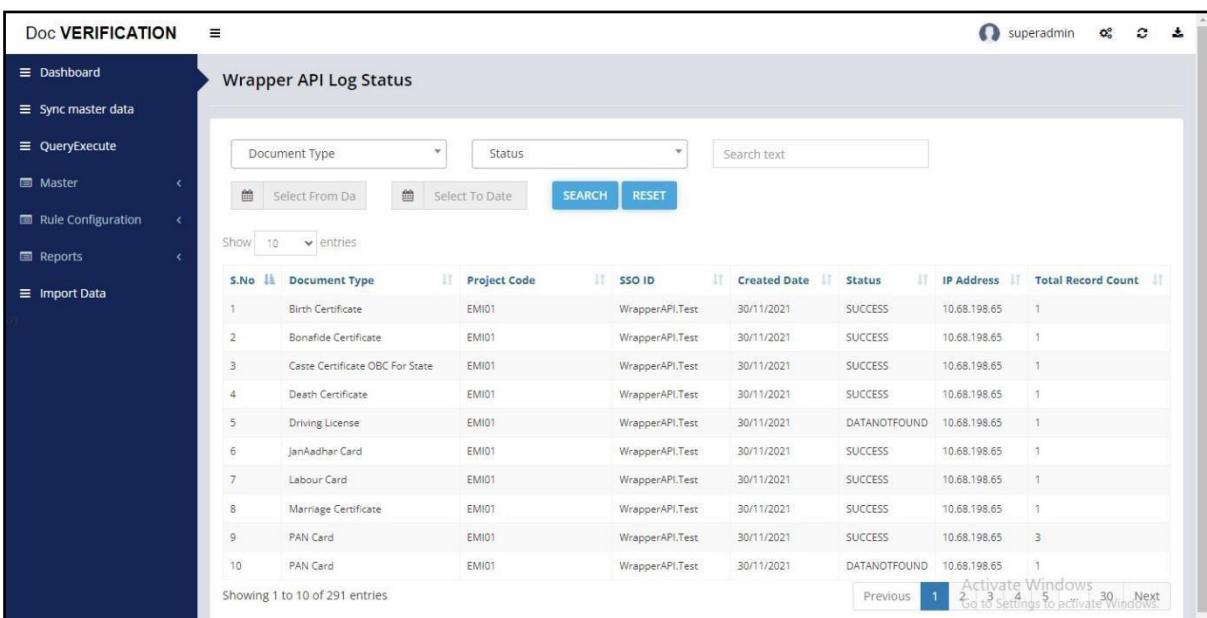


```

else if (model.DocumentTypeId == Convert.ToInt32(DocumentTypeenum.Jamabandi))
{
  if (model.districtId == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter District Id (districtId)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.tehsilId == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Tehsil Id (tehsilId)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.villId == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Village Id (villId)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.khata == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Khata Number (khata)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.UserName == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter User Name (UserName)", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.Password == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Password", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.ApplicantName == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Applicant Name", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.ApplicantMobile == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Applicant Mobile", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
  if (model.ApplicantAddress == null)
  {
    return new CustomActionResult<object>(HttpStatusCodes.BadHttpRequest, new ResponseModel<object>() { Data = null, TotalRecordsCount = 0, isSuccess = false, Message = "Pls enter Applicant Address", DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });
  }
}
ServicePointManager.ServerCertificateValidationCallback = new System.Net.Security.RemoteCertificateValidationCallback(delegate { return true; });
JamabandiServiceClient _ejisnehaarSoapClient = new JamabandiService.ejisnehaarSoapClient();
var result = _ejisnehaarSoapClient.JamabandiService.ValidateDistrict(model.districtId, model.tehsilId, model.villId, model.khata, model.UserName, model.Password, model.ApplicantName, model.ApplicantMobile, model.ApplicantAddress);
SaveWrapperLog(model);
return new CustomActionResult<object>(HttpStatusCodes.Accepted, new ResponseModel<object>() { Data = ROW, TotalRecordsCount = 1, isSuccess = true, Message = CustomMessages.DATARETRIVED, DocumentName = CoreExtensions.GetDocumentTypeDisplayName(model.DocumentTypeId) });

```

4. Log's Detail (Wrapper API's):



S.No	Document Type	Project Code	SSO ID	Created Date	Status	IP Address	Total Record Count
1	Birth Certificate	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
2	Bonafide Certificate	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
3	Caste Certificate OBC For State	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
4	Death Certificate	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
5	Driving License	EMI01	WrapperAPI.Test	30/11/2021	DATANOTFOUND	10.68.198.65	1
6	JanAadhar Card	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
7	Labour Card	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
8	Marriage Certificate	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	1
9	PAN Card	EMI01	WrapperAPI.Test	30/11/2021	SUCCESS	10.68.198.65	3
10	PAN Card	EMI01	WrapperAPI.Test	30/11/2021	DATANOTFOUND	10.68.198.65	1

Showing 1 to 10 of 291 entries

5. Error Codes:

Error Code	Error Description
101	Data Not found
102	Invalid Document Type

Document Verification API's Specification

105

Unauthorized Access

Document Verification API's Specification

6. Contact Details for Integration Support:

S. N	Name	Designation/Role	Mobile Number	E-mail
1	Rama Tiwari	Developer	8955700486	ramatiwari279@gmail.com
2	Tulsa Ram Parihar	Programmer	9460915879	Tulsaparihar.doit@rajasthan.gov.in
3	Ranveer Singh	ACP (Dy. Director)	9784436635	Ranveersingh.doit@rajasthan.gov.in

Note: -

1. The API will run only on internet provided by RajSwan/Rajnet/RajWifi.
2. This is a sample copy so, please input all parameters as required.

End of the Document
