|  |
| --- |
| LIN Only Regime - Shram Suvidha Web APIs V2.2 Blueprint |
| Web Services Specification Document |
| *NIC-STTL-USSP-SWAB-2.2* |

**Abstract:** This requirements document is an attempt to provide an overview of web services required for integration of Shram Suvidha database with various Labour enforcement agencies. The document provides the technical details of Shram Suvidha LIN web services.

**Keywords:** USSP, web service, API, enforcement agencies, Shram Suvidha, Version 2.2, LIN, Labour Identification Number, SSDS

**Prepared by:**



Silver Touch Technologies Ltd.

**Under the Guidance of**

Nantional Informatics Centre (Labour and Employment Division)

*No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of NIC/MoLE.*

**Contributors**

|  |  |
| --- | --- |
| Being Prepared by | Mr. Tarandeep Singh, Project Manager, Silvertouch Technologies Ltd |
| Reviewed by | Sh. Avinash Kumar, Scientist B, NIC |
| Approved by | Sh. Brijesh Krishna Sharma, Scientist D, NIC |

**Amendment log**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Brief Description** | **Section Change** |
| 2.1 | 15/8/2017 | Addition of Update Contact details | 8.1.6, 10.6 |
| 2.2 | 27/10/2017 | Addition of Create Establishment V1.1 | 8.1.5, 10.5 |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Introduction 1](#_Toc485045343)

[2. Intended Audience 1](#_Toc485045344)

[3. Scope 1](#_Toc485045345)

[4. References and Discussions 1](#_Toc485045346)

[4.1. References 1](#_Toc485045347)

[4.2. Discussions 1](#_Toc485045348)

[5. Definition and Abbreviations 2](#_Toc485045349)

[6. Technology Description 3](#_Toc485045350)

[6.1. Shram Suvidha Web Service V2.1 Architecture Description 3](#_Toc485045351)

[6.2. Shram SuvidhA Web Service Framework Components 3](#_Toc485045352)

[7. Technical Flow 4](#_Toc485045353)

[8. Technical Details 4](#_Toc485045354)

[8.1. SSDS API 4](#_Toc485045355)

[8.1.1. GET ‘Lin Profile by Lin No’ 4](#_Toc485045356)

[8.1.2. GET ‘Know Your Lin by Identifier with Few Query Parameter’ 5](#_Toc485045357)

[8.1.3. POST ‘Know Your Lin by Identifier’ 6](#_Toc485045358)

[8.1.4. POST ‘Create Establishment’ 7](#_Toc485045359)

[8.1.5. PUT ‘Update Establishment Representative Profile by Lin No’ 10](#_Toc485045360)

[8.1.6. GET ‘Lin Profile Filter by District Id or State Id’ 12](#_Toc485045361)

[9. Header Details 13](#_Toc485045362)

[10. Server API Example Code 14](#_Toc485045363)

[10.1. Lin Profile By Lin No 14](#_Toc485045364)

[10.2. Know Your Lin By Identifier With Few Query Parameter 17](#_Toc485045365)

[10.3. Know your LIN by identifier 19](#_Toc485045366)

[10.4. Create Establishment 21](#_Toc485045367)

[10.5. LIN Profile update 26](#_Toc485045368)

[10.6. Lin Profile Filter By District Id Or State Id 27](#_Toc485045369)

[Annexure 30](#_Toc485045370)

[A. Message Code 30](#_Toc485045371)

[B. Recommendations 30](#_Toc485045372)

# Introduction

Updated Version of Shram Suvidha Portal is integrated with various external systems like EPFO portal, ESIC portal, eBiz portal, State Labour Department Portal etc. In order to communicate data (post or push) with these external systems Shram Suvidha developers have developed a set of web services.

USSP (Unified Shram Suvidha Portal) system consists of one or more web services and many external systems as web service clients. This specification defines Shram Suvidha APIs for the developed and development of web services. Given such a specification, it is possible to use formal verification techniques to demonstrate that a candidate web service is correct with respect to that specification.

The web service requirements are detailed throughout this specification. Implementations may go beyond what is stated in this document, so may need to perform verification steps after implementation.

# Intended Audience

This document is intended for Shram Suvidha Portal and Enforcement Agency (State and Central Govt.) developers by which they will build the facility to provide or consume the data or perform operations.

# Scope

The scope of this document is to cover the webs services technical details for Establishment and LIN Shram Suvidha APIs. It will detail the web services developed for data transfer between enforcement agencies server and USSP servers.

This specification will be referred as ‘Shram Suvidha Web APIs Blueprint (SWAB)’ and its scope ingress below details:

1. Web service technology description
2. Web service technical flow
3. Web service technical details
4. SSDS API (LIN and Establishment web services)
5. Input and Output Parameters
6. Server API code

# References and Discussions

The following documents, standards, websites and discussion have been consulted/referred when preparing this document,

## References

* System Requirements Specifications for ‘Shram Suvidha Portal V0.3’
* SSDS Web Service Developer Manual
* Web Services developed in Shram Suvidha Version 1.0

## Discussions

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Date** | **Participants** |
| 1. |  |  |
| 2. |  |  |

# Definition and Abbreviations

Please refer below list of terms, definitions and abbreviations related to this document:

|  |  |
| --- | --- |
| **List of Terms (sorted A TO Z)** | |
| **Terms** | **Descriptions** |
| API | Application Programming Interface - defined interfaces or set of services through which interactions happen between applications. |
| CLC | Chief Labour Commissioner |
| DGMS | Directorate General of Mines Safety |
| EA | Enforcement Agency |
| EPFO | Employee's Provident Fund Organization |
| ESIC | Employee's State Insurance Corporation |
| Input | A message that is sent to the server |
| IR | Inspection Report |
| LIN | Labour Identification Number |
| MoLE | Ministry of Labour and Employment |
| NIC | National Informatics Centre |
| Output | A message that is sent to the client |
| REST | Representational State Transfer |
| SSDS | Shram Suvidha Data Sharing |
| SWAB | Shram Suvidha Web APIs Blueprint |
| URI | Uniform Resource Identifier (e.g. http://api.shramsuvidha.gov.in/services/webservices) |
| USSP | Unified Shram Suvidha Portal |
| Web service (Web based API) | It is a service offered by Shram Suvidha, utilized for machine-to-machine (client and server) communication, more specifically for transferring machine readable file formats such as XML and JSON through a URI. |

# Technology Description

## Shram Suvidha Web Service V2.1 Architecture Description

The Shram Suvidha web services are created on **Application Programming Interface (API)** using **CakePHP 2.8,** for message exchange, based on JSON media type Communication between the web service and client. Shram Suvidha web APIs adhere to HTTP-based **RESTful APIs**.

CakePHP is an open-source quality framework for developing RESTful Web Services in PHP. It follows the model–view–controller (MVC) approach and is written in PHP, modeled after the concepts of Ruby on Rails, and distributed under the MIT License.

Web services based on REST (**RE**presentational **S**tate **T**ransfer) Architecture are known as RESTful web services. These web services use HTTP methods to implement the concept of REST architecture. A RESTful web service usually defines a URI, Uniform Resource Identifier a service, provides resource representation such as JSON and set of HTTP Methods like GET, PUT, POST, and DELETE etc.

**HTTP Methods**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Uniform Resource Locator (URL)** | **GET** | **PUT** | **POST** | **DELETE** |
| Collection, such as  <http://api.example.com/resources/>  Member of collection, such as  <http://api.example.com/resources/item17> | List the URIs and perhaps other details of the collection's members or the addressed member of the collection | Replace the collection or addressed member of the collection, or if it does not exist, create it. | Create a new entry in the collection | Delete the collection or addressed member of the collection |

## Shram SuvidhA Web Service Framework Components

The Shram Suvidha web services would be accessed using <http://api.shramsuvidha.gov.in/services/webservice>, other details are:

1. **Webservice Framework:** CakePHP 2.8 + RESTful
2. **Host:** <http://api.shramsuvidha.gov.in>
3. **Base URI:** /services/webservices
4. **Content-Type**: application/json
5. **Content-Encoding**: gzip
6. **Cache-Control**: no-cache

# Technical Flow

Web Service follows a set of operations on received request from sender. Server needs to verify the schema, message size, content, content type and character sets. Filter, scrub, and reject input and output before additional processing.

1. **Validate Source**

Client provides the IP address from headers, server authenticate this IP at the time it receives the file. If server receives file from wrong IP, it generates the system error 9xxx.

1. **Validate Schema**

At this step file structure is parsed and checked for its structure validity. Application would send relevant error messages based on schema invalidity. Error messages for schema validation are defined with 8xxx code series e.g. 8003 extra field received.

1. **Validate Data**

The validation check is performed on file data and reference parameters are checked like reference date should be in correct date format. Error messages should be sent for invalid data defined with 7xxx and 5xxx series like missing mandatory information or invalid values.

1. **Authenticate User**

The server checks the authentication parameter like file security code and user id of the sender. If the parameters are invalid relevant error messages are sent to the sender. Error messages for authentication are defined with 4xxx series like 4001: Invalid User-id/Password.

1. **Process Request**

Request parameters are processed and target location is updated. If any exceptions are found with request parameters server application send relevant error messages defined with 3xxx series.

1. **Generate Response**

Server application generates the response as output in defined message type for the file received. Error messages for response are defined with 1xxx and 2xxx.

1. **Maintain Audit Log**

Details like session id, user id, IP are maintained at the backend. Audit logs enable to keep a complete, persistent record of session, requests, responses, and fault.

# Technical Details

Technical details provide the necessary information for the developed Shram Suvidha web services. The details include web service standards, description, resource information and input/ output parameters from developer’s perspective. Specifications are needed to develop web services in a standard format and avoid errors due to lack of compatibility, for instance, in interoperability issues with external system.

## SSDS API

**LIN and Establishment Services**

### GET ‘Lin Profile by Lin No’

These web service take establishments LIN Number as request parameter and in response provide LIN profile summary.

**Resource Information**

**Service ID**: LP4-GET

**Version**: 2.1

**URI**: <Base URI>/lin/profile/{X-XXXX-XXXX-X}

**Method**: HTTPS/1.1 GET

**Parameters**

1. **Input parameter**
2. Request Parameter
   1. Lin No (Data Type=string, data length=13)
3. **Output/Response Parameters**
4. Message
   1. Message Code (Data Type=Integer, data length=4)
   2. Message (actual message) (Data Type=string, data length=255)
5. LIN Profile Type
   * 1. Name (Data Type=string, data length=255)
     2. *Address*
        1. Premise No./Name, Sub-Locality/Colony (Data Type=string, data length=100)
        2. Locality/Area/Suburb (Data Type=string, data length=100)
        3. Town/City (Data Type=string, data length=50)
        4. Pin Code (Data Type=Integer, data length=6)
        5. State Code (Data Type=string, data length=2)
        6. District Code (Data Type=string, data length=3)
        7. Country (Data Type=string, data length=50)
     3. *Telephone/Email Contacts (Multiple)*
        1. Contact Type like Mobile, Landline, Email, Website etc. (Data Type=string, data length=50)
        2. Contact Value (Data Type=string, data length=255)
     4. *Contact Persons (Multiple)*
        1. Name (Data Type=string, data length=255)
        2. Address (Data Type=string, data length=255)
        3. Office and Role like Owner, Partner, and Authorised Signatory etc. (Data Type=string, data length=50)
     5. Set-up Date (Data Type=date, date format = dd/mm/yyyy)
     6. Sector like Govt., Private, Public Limited, Co-Op Society (Data Type=string, data length=50)
     7. Category (Data Type=string, data length=255)
     8. Involved in Hazardous activity? Yes/No (Data Type=string, data length=3)
     9. Involved in Dangerous activity? Yes/No (Data Type=string, data length=3)
     10. Seasonal (Yes/No) (Data Type=string, data length=3)
     11. Total Number of Employees at the time of LIN Allotment (Data Type=string, data length=10)
     12. Nature of Activity (Data Type=string, data length=255)
     13. Brief Description (Data Type=string, data length=1000)
     14. Identifiers (Multiple)
         1. Identifier Type like EPFO-REGN, ESIC-REGN, PAN etc. (Data Type=string, data length=50)
         2. Identifier Value (Data Type=string, data length=255)
     15. Entity Relationship (Multiple)
         1. Related Entity LIN (Data Type=string, data length=13)
         2. Related Entity Name (Data Type=string, data length=255)
         3. Relationship like Head Office, Branch Office etc. (Data Type=string, data length=50)

For web service code refer [server API example code](#_LinProfileByLinNo).

### GET ‘Know Your Lin by Identifier with Few Query Parameter’

This web service take establishment identifier details as request parameter and in response provide LIN numbers and LIN profile summary.

**Resource Information**

**Service ID**: KYL7-GET

**Version**: 2.1

**URI**: <Base URI>/lin/knowyourlin/identifier?acronym=PAN&identifier=XXXXXXXXXX

**Method**: HTTPS/1.1 GET

**Parameters**

1. **Input parameter**
2. Request Parameter
   1. Identifier Acronym (Data Type=string, data length=10)
   2. Identifier Value (Data Type=string, data length=100)
3. **Output/Response Parameters**
4. Message
   1. Message Code (Data Type=Integer, data length=4)
   2. Message (actual message) (Data Type=string, data length=255)
5. LIN Profile
   1. Establishment Name (Data Type=string, data length=255)
   2. LIN (Data Type=string, data length=13)
   3. *Address*
      * 1. Premise No./Name, Sub-Locality/Colony (Data Type=string, data length=100)
        2. Locality/Area/Suburb (Data Type=string, data length=100)
        3. Town/City (Data Type=string, data length=50)
        4. Pin Code (Data Type=Integer, data length=6)
        5. State Code (Data Type=string, data length=2)
        6. District Code (Data Type=string, data length=3)
        7. Country (Data Type=string, data length=50)
   4. *Telephone/Email Contacts (Multiple)*
      1. Contact Type like Mobile, Landline, Email, Website etc. (Data Type=string, data length=50)
      2. Contact Value (Data Type=string, data length=255)
   5. *Contact Persons (Multiple)*
      1. Name (Data Type=string, data length=255)
      2. Office and Role like Owner, Partner, and Authorised Signatory etc. (Data Type=string, data length=50)

For webservice code refer [server API example code](#_KnowYourLinByIdentifierWithFewQuery).

### POST ‘Know Your Lin by Identifier’

Take establishment identifier details as input and in response provide LIN numbers with corresponding establishment name. This webservice is open URI and can be accessed anywhere.

**Resource Information**

**Service ID**: KYL8-POST

**Version**: 2.0

**URI**: <Base URI>/lin/knowyourlin/identifier

**Method**: HTTPS/1.1 POST

Authorization: User Authentication inside Service - accept encoded Password using Base 64 Algorithm

Cache-Control: cached at server end

**Parameters**

1. **Input parameter**
2. Reference Parameter
   1. Office Acronym (Data Type=string, data length=10)
   2. Reference Number (Data Type=string, data length=10)
   3. Reference date (Data Type=date, format = dd/mm/yyyy)
3. Authentication Parameter
   1. User Id (Data Type=string, data length=20)
   2. Password (Data Type=string, data length=50)
4. Request Parameter
   1. Identifier Acronym (Data Type=string, data length=10)
   2. Identifier Value (Data Type=string, data length=100)
5. **Output/Response Parameters**
6. Message Type
7. Message
8. Message Code (Data Type=Integer, data length=4)
9. Message (actual message) (Data Type=string, data length=255)
10. LIN Profile
11. Establishment Name (Data Type=string, data length=255)
12. Address
    1. Premise No./Name, Sub-Locality/Colony (Data Type=string, data length=100)
    2. Locality/Area/Suburb (Data Type=string, data length=100)
    3. Town/City (Data Type=string, data length=50)
    4. District (Data Type=string, data length=50)
    5. Pin Code (Data Type=Integer, data length=6)
    6. State (Data Type=string, data length=50)
    7. Country (Data Type=string, data length=50)
13. LIN

For webservice code refer [server API example code](#_Know_your_LIN).

### POST ‘Create Establishment’ (Version 1.0)

Takes establishment details or profile as input and provides Service Request Number (SRN) in response.

**Resource Information**

**Service ID**: CE10-POST

**Version**: 2.0

**URI**: <Base URI>/establishment/create

**Method**: HTTPS/1.1 POST

Authorization: User Authentication inside Service - accept encoded Password using Base 64 Algorithm

Cache-Control: cached at server end

**Parameters**

1. **Input parameter**
2. Authentication Parameter
   1. User Id (Data Type = string, data length = 20)
   2. Password (Data Type = string, data length = 50))
3. Reference Parameter
   1. Office Acronym (Data Type = string, data length = 10)
   2. Reference Number (Data Type = integer, data length = 10)
   3. Reference Date (Data Type = date, Date format = dd/mm/yyyy)
4. Request Parameter
5. Establishment Name (Data Type=string, data length=255)
6. Establishment Address
   * 1. Premise No./Name, (Data Type=string, data length=100)
     2. Sub-Locality/Colony (Data Type=string, data length=100)
     3. Locality/Area/Suburb (Data Type=string, data length=100)
     4. State (Data Type=string, data length=50)
     5. District (Data Type=string, data length=50)
     6. Pincode (Data Type=Integer, data length=6)
     7. Country (Data Type=string, data length=50)
7. Contacts
8. Phone Mobiles(Multiple)
   1. Mobile 1 (Data Type=Integer, data length=10)
   2. Mobile 2 (Data Type=Integer, data length=10)
9. Phone Landlines (Multiple)
   1. Landline 1 (Data Type=Integer, data length=11)
   2. Landline 2 (Data Type=Integer, data length=11)
10. Emails (Multiple)
    1. Email 1 (Data Type=string, data length=100)
    2. Email 2 (Data Type=string, data length=100)
11. Contact Person (Multiple)
12. Person Name (Data Type=string, data length=100)
13. Role/Designation (Data Type=string, data length=50)
14. Email (Data Type=string, data length=100)
15. Phone (Data Type=Integer, data length=10)
16. Acts Applicable (Multiple)
    1. Act Acronym (Data Type=string, data length=10)
    2. Coverage Date (Optional) (Data Type=date, Date format = dd/mm/yyyy)
17. Identifiers (Multiple)
18. Identifier Type Acronym (Data Type=string, data length=10)
19. Identifier Value (Data Type=string, data length=100)
20. Total number of Workers
21. Contract Male Workers (Data Type=Integer, data length=6)
22. Contract Female Workers (Data Type=Integer, data length=6)
23. Regular Male Workers (Data Type=Integer, data length=6)
24. Regular Female Workers (Data Type=Integer, data length=6)
25. Setup Date (Data Type=Date, Date format = dd/mm/yyyy)
    * 1. Involved in Hazardous activity? Yes/No (Optional) (Data Type=String, data length=3)
      2. Involved in Dangerous activity? Yes/No (Optional) Yes/No (Optional) (Data Type=String, data length=3)
      3. Seasonal (Yes/No) (Optional) Yes/No (Optional) (Data Type=String, data length=3)
      4. Sector Acronym Yes/No (Optional) (Data Type=String, data length=50)
      5. Category Acronym Yes/No (Optional) (Data Type=String, data length=50)
      6. NIC Code (5Dgit Numeric Code) (Data Type=Integer, data length=5)
26. **Output/Response Parameters (Here there are 3 Cases of Response)**

**>In Case of De-Duplication score indicates high probability of Establishment Already in database**

1. Message Type
2. Message
   * 1. Message Code Yes/No (Optional)
3. Message (actual message) (Data Type=String, data length=255)
4. Reference Number (Data Type=Integer, data length=10)
5. Response
6. LIN number for LIN seeding (Data Type=String, data length=13)
7. Establishment Creation Message (Data Type=String, data length=255)

**>In Case of De-Duplication score indicates high probability of Establishment data as unique**

1. Message Type
2. Message
3. Message Code (Data Type=Integer, data length=4)
4. Message (actual message) (Data Type=String, data length=255)
5. Reference Number (Data Type=Integer, data length=10)
6. Response
7. New LIN number generated for Establishment (Data Type=String, data length=13)
8. Establishment Creation Message (Data Type=String, data length=255)

**>In Case of De-Duplication score indicates that Manual Intervention needed for LIN generation as per establishment data**

1. Message Type
2. Message
3. Message Code (Data Type=Integer, data length=4)
4. Message (actual message) (Data Type=String, data length=255)
5. Reference Number (Data Type=Integer, data length=10)
6. Response
7. Top 5 Matching Result
   1. Establishment 1
      1. Establishment Name (Data Type=String, data length=255)
      2. LIN (Data Type=String, data length=13)
      3. Locality/Area/Suburb, District, State (Data Type=String, data length=100)
      4. Mobile Number (Data Type=Integer, data length=10)
      5. Email Id (Data Type=String, data length=100)
   2. Establishment 2
      1. Establishment Name (Data Type=String, data length=255)
      2. LIN (Data Type=String, data length=13)
      3. Locality/Area/Suburb, District, State (Data Type=String, data length=100)
      4. Mobile Number (Data Type=Integer, data length=10)
      5. Email Id (Data Type=String, data length=100)
   3. Establishment 3
      1. Establishment Name (Data Type=String, data length=255)
      2. LIN (Data Type=String, data length=13)
      3. Locality/Area/Suburb, District, State (Data Type=String, data length=100)
      4. Mobile Number (Data Type=Integer, data length=10)
      5. Email Id (Data Type=String, data length=50)
   4. Establishment 4
      1. Establishment Name (Data Type=String, data length=255)
      2. LIN (Data Type=String, data length=13)
      3. Locality/Area/Suburb, District, State (Data Type=String, data length=100)
      4. Mobile Number (Data Type=Integer, data length=10)
      5. Email Id (Data Type=String, data length=100)
   5. Establishment 5
      1. Establishment Name (Data Type=String, data length=255)
      2. LIN (Data Type=String, data length=13)
      3. Locality/Area/Suburb, District, State (Data Type=String, data length=100)
      4. Mobile Number (Data Type=Integer, data length=10)
      5. Email Id (Data Type=String, data length=100)
8. Result Message
9. **Input/Request Parameters (For 3rd Case above in Output/Response Parameters)**

**>In Case, requester select one particular LIN as matching establishment out of 5 results**

* 1. Authentication Parameter
     1. Userid (Data Type=String, data length=20)
     2. Password (Data Type=String, data length=50)
  2. Reference Parameter
     1. OfficeAcronym (Data Type=String, data length=10)
     2. referenceNumber (Data Type=Integer, data length=10)
     3. referenceDate (Data Type=Date, Date format = dd/mm/yyyy)
  3. Request Parameter
     1. LIN Number seeded (Data Type=String, data length=13)

**>In Case, requester communicate to generate New LIN for establishment as top 5 results do not match**

1. Authentication Parameter
   * 1. Userid (Data Type=String, data length=20)
     2. Password (Data Type=String, data length=50)
2. Reference Parameter
   * 1. OfficeAcronym (Data Type=String, data length=10)
     2. referenceNumber (Data Type=Integer, data length=10)
     3. referenceDate (Data Type=Date, Date format = dd/mm/yyyy)
3. Request Parameter
   * 1. Generate New LIN (Data Type=String, data length=13)
4. **Output/Response Parameters (For 2rd Case above in Input/Request Parameters)**
5. Message Type
6. Message
7. Message Code (Data Type=String, data length=4)
8. Message (actual message) (Data Type=String, data length=255)
9. Reference Number (Data Type=Integer, data length=10)
10. Response
11. New LIN number generated for Establishment (Data Type=String, data length=13)
12. Establishment Creation Message (Data Type=String, data length=255)

For webservice code refer [server API example code](#_CreateEstablishment).

### POST ‘Create Establishment’ (Version 1.1)

This service has been constructed for State Enforcement Agencies, So, that it be convenient for the system to Generate statements that are to be sent to Labour Bureau. It’s up to agencies choice to which version they want to integrate to serve their purpose, same as V1.0 it takes establishment details or profile as input and provides Service Request Number (SRN) in response.

**Resource Information**

**Service ID**: CE10-POST

**Version**: 2.0

**URI**: <Base URI>/establishment/create

**Method**: HTTPS/1.1 POST

Authorization: User Authentication inside Service - accept encoded Password using Base 64 Algorithm

Cache-Control: cached at server end

**Parameters**

1. **Input parameter**
2. Authentication Parameter
   1. User Id (Data Type = string, data length = 20)
   2. Password (Data Type = string, data length = 50))
3. Reference Parameter
   1. Office Acronym (Data Type = string, data length = 10)
   2. Reference Number (Data Type = integer, data length = 10)
   3. Reference Date (Data Type = date, Date format = dd/mm/yyyy)
4. Request Parameter
5. Establishment Name (Data Type=string, data length=255)
6. Establishment Address
7. Premise No./Name, (Data Type=string, data length=100)
8. Sub-Locality/Colony (Data Type=string, data length=100)
9. Locality/Area/Suburb (Data Type=string, data length=100)
10. State (Data Type=string, data length=50)
11. District (Data Type=string, data length=50)
12. Pincode (Data Type=Integer, data length=6)
13. Country (Data Type=string, data length=50)
14. Contacts
15. Phone Mobiles(Multiple)
    1. Mobile 1 (Data Type=Integer, data length=10)
    2. Mobile 2 (Data Type=Integer, data length=10)
16. Phone Landlines (Multiple)
    1. Landline 1 (Data Type=Integer, data length=11)
    2. Landline 2 (Data Type=Integer, data length=11)
17. Emails (Multiple)
    1. Email 1 (Data Type=string, data length=100)
    2. Email 2 (Data Type=string, data length=100)
18. Owner/Employer/Occupier Details
    * + 1. Designation (Data Type=string, data length=50)
        2. Name (Data Type=string, data length=50)
        3. Address
           1. Premise No./Name, (Data Type=string, data length=100)
           2. Sub-Locality/Colony (Data Type=string, data length=100)
           3. Locality/Area/Suburb (Data Type=string, data length=100)
           4. State (Data Type=string, data length=50)
           5. District (Data Type=string, data length=50)
           6. Pincode (Data Type=Integer, data length=6)
           7. Country (Data Type=string, data length=50)
        4. Mobile (Data Type=Integer, data length=10)
        5. Landline (Data Type=Integer, data length=11)
        6. Email (Data Type=string, data length=100)
19. Contact Person (Multiple)
20. Person Name (Data Type=string, data length=100)
21. Role/Designation (Data Type=string, data length=50)
22. Email (Data Type=string, data length=100)
23. Phone (Data Type=Integer, data length=10)
24. Acts Applicable (Multiple)
25. Act Acronym (Data Type=string, data length=10)
26. Coverage Date (Optional) (Data Type=date, Date format = dd/mm/yyyy)
27. Identifiers (Multiple)
28. Identifier Type Acronym (Data Type=string, data length=10)
29. Identifier Value (Data Type=string, data length=100)
30. Employer Registrations
31. Employer registration under the act (Data Type=string, data length=10)
32. Registration No (Data Type=string, data length=20)
33. Registration Date (Data Type=Date, Date format = dd/mm/yyyy)
34. No. of Workers at the time of registration (Data Type=Integer, data length=6)
35. Date of Renewal of Registration (Data Type=Date, Date format = dd/mm/yyyy)
36. No. of workers at the time of renewal of registration (Data Type=Integer, data length=6)
37. License No (Data Type=string, data length=20)
38. License Date (Data Type=Date, Date format = dd/mm/yyyy)
39. Date of Renewal of License (Data Type=Date, Date format = dd/mm/yyyy)
40. No. of workers at the time of renewal of license (Data Type=Integer, data length=6)
41. Total number of Workers as on today
42. Contract Male Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
43. Contract Female Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
44. Regular Male Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
45. Regular Female Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
      1. Total Number of Workers as on today, drawing wages Rs. 24000/- or Less
46. Contract Male Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
47. Contract Female Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
48. Regular Male Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
49. Regular Female Workers
    * + - 1. Highly Skilled (Data Type=Integer, data length=6)
          2. Skilled (Data Type=Integer, data length=6)
          3. Semi-Skilled (Data Type=Integer, data length=6)
          4. Unskilled (Data Type=Integer, data length=6)
50. Setup Date (Data Type=Date, Date format = dd/mm/yyyy)
    * 1. Establishment Working Status- Working/Closed (Optional) (Data Type=String, data length=10)
      2. If Factory, Registration under Section (2mi/2mii/Sec 85) Data Type=String, data length=10)
      3. If Factory, Power HP (Data Type=String, data length=10)
      4. Do establishment Notified for appointment of safety officer? Yes/No (Optional) (Data Type=String, data length=3)
      5. Involved in Hazardous activity? Yes/No (Optional) (Data Type=String, data length=3)
      6. Involved in Dangerous activity? Yes/No (Optional) Yes/No (Optional) (Data Type=String, data length=3)
      7. Seasonal (Yes/No) (Optional) Yes/No (Optional) (Data Type=String, data length=3)
      8. Sector Acronym Yes/No (Optional) (Data Type=String, data length=50)
      9. Category Acronym Yes/No (Optional) (Data Type=String, data length=50)
      10. NIC Code (5Dgit Numeric Code) (Data Type=Integer, data length=5)
      11. Primary Business Activity (Data Type=String, data length=50)
      12. Nature of Work (Data Type=String, data length=50)
      13. Sub Category of Nature of Work (Data Type=String, data length=50)
      14. Brief Description of Work (Data Type=String, data length=150)
51. **Output/Response Parameters (Response will be same as in Version 1.0)**

### PUT ‘Update Establishment Representative Profile by Lin No’

This USSP web service aims to update the establishment representative profile only by the source enforcement agency. The service has input request parameters as LIN No., establishment representative name (First, middle and last), email id, contact number and address. The service in response provides the success or failure message.

This services uses the put operation, in this case, if establishment details requested to update doesn’t exists in system, it creates the new entry for them.

**Validations**

1. **Validate IP Source**

Client provides the IP address from headers, server authenticate this IP at the time it receives the file. If server receives file from wrong IP, it generates the system error 9xxx. Server finds out the enforcement agency corresponding to the request IP in database.

1. **Validate LIN Source Agency**

Server finds out the source enforcement agency (IP) corresponding to the requested LIN in database. If requested IP enforcement agency and LIN source agency are same system allows update Establishment Representative Profile operation.

**Resource Information**

**Service Name:** update establishment representative

**Service ID**: ERP19-PUT

**Version**: 2.0

**URI**: Host /<Base URI>/establishment/representative/update

**Method**: HTTPS/1.1 PUT

**Parameters**

1. **Input parameter**
2. Authentication Parameter
   1. userName (mandatory) (Data Type=String, data length=20)
   2. password (mandatory) (Data Type=String, data length=50)
3. Reference Parameter
4. referenceNumber (mandatory) (Data Type=Integer, data length=10)
5. referenceDate (mandatory) (Data Type=Date, Date format = dd/mm/yyyy)
6. Request Parameter
   1. Lin No (mandatory) (Data Type=String, data length=13)
   2. Establishment Representative
      1. First Name (mandatory) (Data Type=String, data length=50)
      2. Middle Name (Data Type=String, data length=50)
      3. Last Name (Data Type=String, data length=50)
      4. Email Id (mandatory) (Data Type=String, data length=100)
      5. Mobile Number (mandatory) (Data Type=Integer, data length=10)
      6. Address
         1. Premise No./Name, Sub-Locality/Colony (mandatory) (Data Type=String, data length=100)
         2. Locality/Area/Suburb (Data Type=String, data length=100)
         3. stateCode (Agency can get same from master state webservice) (Data Type=String, data length=3)
         4. districtCode (Agency can get same from master district webservice) (Data Type=String, data length=3)
         5. Pincode (Data Type=Integer, data length=6)
         6. Country (default - India) (Data Type=String, data length=50)
7. **Output/Response Parameters**
8. Message Type
9. Message
10. Message Code (Data Type=String, data length=4)
11. Message (actual message) (Data Type=String, data length=255)
    1. Response Message
12. Establishment Representative Data Updated Successfully
13. Source Agency mismatch with request LIN.
14. Requested LIN not found.

### GET ‘Lin Profile Filter by District Id or State Id’

Takes LIN number as input and generate response as LIN profile with district name and state name.

**Resource Information**

**Service ID**: LP11-GET

**Version**: 2.0

**URI**: <Base URI>/lin/profile/filter/{eitherDistrictOrStateCode}

**Method**: HTTPS/1.1 GET

**Parameters**

1. **Input parameter**
2. Request Parameter
   1. LinNo (Data Type=String, data length=13)
3. **Output/Response Parameters**
4. Message Type
5. Message
6. Message Code (Data Type=Integer, data length=4)
7. Message (actual message) (Data Type=String, data length=255)
8. Request Information
   1. CodeProvided (Data Type=String, data length=50)
   2. DistrictCode (Data Type=String, data length=3)
   3. StateCode (Data Type=String, data length=2)
   4. DistrictName (Data Type=String, data length=50)
   5. StateName (Data Type=String, data length=50)
9. Lin Profile
   * 1. Name (Data Type=String, data length=255)
     2. *Address* (Data Type=String, data length=100)
        1. Premise No./Name, Sub-Locality/Colony (Data Type=String, data length=100)
        2. Locality/Area/Suburb (Data Type=String, data length=100)
        3. Town/City (Data Type=String, data length=50)
        4. Pin Code (Data Type=Integer, data length=6)
        5. State Code (Data Type=String, data length=.)
        6. District Code (Data Type=String, data length=3)
        7. Country (Data Type=String, data length=50)
     3. *Telephone/Email Contacts (Multiple)*
        1. Contact Type like Mobile, Landline, Email, Website etc. (Data Type=String, data length=50)
        2. Contact Value (Data Type=String, data length=255)
     4. *Contact Persons (Multiple)*
        1. Name (Data Type=String, data length=100)
        2. Address (Data Type=String, data length=255)
        3. Office and Role like Owner, Partner, and Authorised Signatory etc. (Data Type=String, data length=50)
     5. Set-up Date (Data Type=Date)
     6. Sector like Govt., Private, Public Limited, Co-Op Society (Data Type=String, data length=50)
     7. Category (Data Type=String, data length=100)
     8. Involved in Hazardous activity? Yes/No (Data Type=String, data length=3)
     9. Involved in Dangerous activity? Yes/No (Data Type=String, data length=3)
     10. Seasonal (Yes/No) (Data Type=String, data length=3)
     11. Total Number of Employees at the time of LIN Allotment (Data Type=String, data length=6)
     12. Nature of Activity (Data Type=String, data length=100)
     13. Brief Description (Data Type=String, data length=1000)
     14. Identifiers (Multiple)
         1. Identifier Type like EPFO-REGN, ESIC-REGN, PAN etc. (Data Type=String, data length=50)
         2. Identifier Value (Data Type=String, data length=100)
     15. Entity Relationship (Multiple)
         1. Related Entity LIN (Data Type=String, data length=13)
         2. Related Entity Name (Data Type=String, data length=255)
         3. Relationship like Head Office, Branch Office etc. (Data Type=String, data length=50)

For webservice code refer [server API example code](#_LinProfileFilterByDistrictIdOrState).

# Header Details

RESTful web service **header** is read only and it returns only HTTP Header, no Body. It defines the operating parameters of the transaction, the so called metadata. All fields are name-value pairs.

|  |
| --- |
| Access-Control-Allow-Credentials → true  Access-Control-Allow-Headers → origin, content-type, accept, authorization,X-Requested-With  Access-Control-Allow-Methods → GET, POST, PUT, DELETE, OPTIONS, HEAD  Access-Control-Allow-Origin →  Access-Control-Allow-Origin  Specifies a URI that may access the resource. For requests without credentials, the server may specify '\*' as a wildcard, thereby allowing any origin to access the resource.  \*  Access-Control-Max-Age → 3600  Content-Encoding → gzip  Content-Length → 1652  Content-Type → application/json  Date → Thu, 10 Sep 2015 04:03:05 GMT  Server → Apache-Coyote/1.1  Web-Serivces-Powered-By → NIC-INDIA - Shram Suvidha - Unified Portal for Labour and Employment |

# Server API Example Code

## Lin Profile By Lin No

Following is the sample data for ‘**LIN Profile by LIN No**’ web service:

**Example Request**

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET (LP4-GET)** | http://api.shramsuvidha.gov.in/services/webservice/lin/profile/{LIN} |

|  |
| --- |
|  |

**Request**

|  |
| --- |
| http://api.shramsuvidha.gov.in/services/webservice/lin/profile/1-7629-0699-2 |

**Response**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "LINPROFILEBYLIN",

"serviceName": "LIN Profile by LIN",

"method": "GET",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"lin": "1-5383-0496-5"

},

"message": {

"messageCode": "1001",

"messageText": "Success"

},

"linProfile": {

"establishmentName": "",

"address": {

"premiseNoName": "",

"SubLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"contacts": [

{

"contactType": "",

"conatactValue": ""

},

{

"contactType": "",

"conatactValue": ""

}

],

"contactPersons": [

{

"name": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"roleRepresentation": ""

},

{

"name": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"roleRepresentation": ""

}

],

"setupDate": "",

"sector": "",

"category": "",

"involvedInHazardousActivity": "",

"involvedInDangerousActivity": "",

"seasonal": "",

"totalNumberOfEmployeesAtTheTimeOfLINAllotment": "",

"natureOfActivity": "",

"briefDescription": "",

"identifiers": [

{

"identifierType": "",

"identifierValue": ""

},

{

"identifierType": "",

"identifierValue": ""

}

],

"entityRelationship": [

{

"relatedEntityLIN": "",

"entityName": "",

"relationship": ""

},

{

"relatedEntityLIN": "",

"entityName": "",

"relationship": ""

}

]

}

}

}

}

**Error Message**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version":"2.1",

"serviceId":"LINPROFILEBYLIN",

"serviceName":"LIN Profile by LIN",

"method":"GET",

"developedBy":"NIC Labour & Employment Division"

},

"requestParameter": {

"lin":"1-0000-0000-1"

},

"errorMessage": {

"messageCode":"3001",

"messageText":"No Information available about the requested parameter"

}

}

}

}

## Know Your Lin By Identifier With Few Query Parameter

Following is the sample data for ‘**Know your LIN by identifier with few Query Parameters**’ web service:

**Example Request**

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET (KYL7-GET)** | http://api.shramsuvidha.gov.in/services/webservice/lin/knowyourlin/identifier?acronym={Issuer Acronym}&identifier={Identifier Value} |

|  |
| --- |
|  |

**Request**

|  |
| --- |
| http://api.shramsuvidha.gov.in/services/webservice/ lin/knowyourlin/identifier?acronym=ESIC-REGN&identifier=34000260630001101 |

**Response**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "KNOWYOURLINFEWQUERY",

"serviceName": "Know Your LIN By Identifier with Few Query Parameters",

"method": "GET",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"identifierAcronym": "ESIC-REGN"

,"identifier Value":"34000260630001101"},

"message": {

"messageCode": "1001",

"messageText": "Success"

},

"linProfile": {

"lin":"X-XXXX-XXXX-X","establishmentName": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"contacts": [

{

"contactType": "",

"conatactValue": ""

},

{

"contactType": "",

"conatactValue": ""

}

],

"contactPersons": [

{

"name": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"roleRepresentation": ""

},

{

"name": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"roleRepresentation": ""

}

]}

}

}

}

**Error Message**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "KNOWYOURLINFEWQUERY",

"serviceName": "Know Your LIN By Identifier with Few Query Parameters",

"method": "GET",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"identifierAcronym": "ESIC-REGN"

,"identifierValue":"34000260630001101"},

"errorMessage": {

"messageCode": "3001",

"messageText": "No Information available about the requested parameter"

}}

}

}

## Know your LIN by identifier

Following is the sample data for ‘**Know your LIN by identifier**’ web service:

**Example Request**

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST (KYL8-POST)** | http://api.shramsuvidha.gov.in/services/webservice/lin/knowyourlin/identifier |

|  |
| --- |
|  |

**Request**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "KNOWYOURLINIDENTIFIER",

"serviceName": "Know Your LIN By Identifier",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},"referenceParameters":{"officeAcronym":"ESICNDLS","referenceNumber":"12589","referenceDate":"12-5-2017"},"authenticationParameters":{"userID":"","password":""},

"requestParameter": {

"identifierAcronym": "ESIC-REGN"

,"identifierValue":"34000260630001101"}}

}

}

**Response**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "KNOWYOURLINIDENTIFIER",

"serviceName": "Know Your LIN By Identifier",

"method": "Post",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"identifierAcronym": "ESIC-REGN"

,"identifierValue":"34000260630001101"},

"message": {

"messageCode": "1001",

"messageText": "Success"

},

"linProfile": {

"establishmentName": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},"lin":"X-XXXX-XXXX-X"}

}

}

}

**Error Message**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "KNOWYOURLINIDENTIFIER",

"serviceName": "Know Your LIN By Identifier",

"method": "Post",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"identifierAcronym": "ESIC-REGN"

,"identifierValue":"34000260630001101"},

"errorMessage": {

"messageCode": "3001",

"messageText": "No Information available about the requested parameter"

}}

}

}

## Create Establishment Version 1.0

Following is the sample data for ‘**Create Establishment**’ web service:

**Example Request**

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST (CE10-POST)** | http://api.shramsuvidha.gov.in/services/webservice/establishment/create |

|  |
| --- |
|  |

**Request**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},

"authenticationParameters": {

"userId": "",

"password": ""

},

"requestParameters": {

"establishmentName": "",

"address": {

"premiseNoName": "",

"subLocalityColonyName": "",

"localityAreaSuburb": "",

"stateCode": "",

"districtCode": "",

"pinCode": "",

"country": ""

},

"contacts": {

"mobiles": [

{

"mobile1": "",

"mobile2": ""

}

],

"landlines": [

{

"landline1": "",

"landline2": ""

}

],

"emails": [

{

"email1": "",

"email2": ""

}

]

},

"contactPerson": [

{

"personName": "",

"roleDesignation": "",

"email": "",

"phone": ""

},

{

"personName": "",

"roledesignation": "",

"email": "",

"phone": ""

}

],

"actsApplicable": {

"acts": [

{

"actAcronym": "",

"coverageDate": "23-11-1999"

},

{

"actAcronym": "",

"coverageDate": "23-11-1999"

}

]

},

"identifiers": [

{

"identifierTypeAcronym": "",

"identifierValue": ""

},

{

"identifierTypeAcronym": "",

"identifierValue": ""

}

],

"totalNumberOfWorkers": {

"contractMaleWorkers": "",

"contractFemaleWorkers": "",

"regularMaleWorkers": "",

"regularFemaleWorkers": ""

},

"setup": "23-11-1999",

"isInvolvedInHazardousActivity": "",

"isInvolvedInDangerousActivity": "",

"isSeasonal": "",

"sectorAcronym": "",

"categoryAcronym": "",

"nicCode": ""

}

}

}

}

**Response in 1st Case of De-Duplication score indicates high probability of Establishment data as unique**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1985"

},"message":{"messageCode":"1002","messageText":"LIN Generated Successfully"},"responseParameters":{"lin":"X-XXXX-XXXX-X","creationMessage":"Establishment Data found unique and New LIN generated"}}

}

}

**Response in 2nd Case of De-Duplication score indicates high probability of Establishment data already exist**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1985"

},"message":{"messageCode":"1003","messageText":"LIN Already Exist"},"responseParameters":{"linSeeded":"X-XXXX-XXXX-X","creationMessage":"Establishment Data already exist in LIN database"}}

}

}

**Response in 3rd Case of De-Duplication score indicates that Manual Intervention needed for LIN generation as per establishment data**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},"message":{"messageCode":"1003","messageText":"Need Your Intervention for LIN generation or Seeding"},"responseParameters":{"top5MatchingEstablishments":[{"establishmentName":"","lin":"","localityAreaSuburbDistrictState":"","mobile":"","email":""},{"establishmentName":"","lin":"","localityAreaSuburbDistrictState":"","mobile":"","email":""},{"establishmentName":"","lin":"","localityAreaSuburbDistrictState":"","mobile":"","email":""},{"establishmentName":"","lin":"","localityAreaSuburbDistrictState":"","mobile":"","email":""},{"establishmentName":"","lin":"","localityAreaSuburbDistrictState":"","mobile":"","email":""}],"creationMessage":"De-Duplication result as per Establishment data sent need manual intervention for Establishment LIN generation/seeding"}}

}

}

**Input/Request Parameters (For 3rd Case above in Output/Response Parameters)**

**Request in Case, requester select one particular LIN as matching establishment out of 5 results**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},

"authenticationParameters": {

"userId": "",

"password": ""

},

"requestParameters": {

"linSelected": "X-XXXX-XXXX-X","result":"Found Matching LIN"}

}

}

}

**Request in Case, requester communicate to generate New LIN for establishment as top 5 results do not match**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},

"authenticationParameters": {

"userId": "",

"password": ""

},

"requestParameters": {

"result":"No Matching LIN in Top 5 Result"}

}

}

}

**Response in Case, requester communicate to generate New LIN for establishment**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},"message":{"messageCode":"1004","messageText":"New LIN generated as per your Intervention"},"responseParameters":{"linGenerated":"X-XXXX-XXXX-X","creationMessage":"Generated New LIN for Establishment as no Matching result found in Top 5 Matching data"}}

}

}

**On validation Error**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},"errorMessage":{"messageCode":"8002","messageText":"Field name mismatch"}}

}

}

## Create Establishment Version 1.1

Following is the sample data for ‘**Create Establishment**’ web service:

**Example Request**

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST (CE10-POST)** | http://api.shramsuvidha.gov.in/services/webservice/establishment/create/V1\_1 |

|  |
| --- |
|  |

**Request**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "CREATEESTABLISHMENT",

"serviceName": "Create Establishment",

"method": "POST",

"developedBy": "NIC Labour & Employment Division"

},

"referenceParameters": {

"officeAcronym": "ESICNDLS",

"referenceNumber": "12589",

"referenceDate": "23-11-1999"

},

"authenticationParameters": {

"userId": "",

"password": ""

},

"requestParameters": {

"establishmentName": "",

"address": {

"premiseNoName": "",

"subLocalityColonyName": "",

"localityAreaSuburb": "",

"stateCode": "",

"districtCode": "",

"pinCode": "",

"country": ""

},

"contacts": {

"mobiles": [

{

"mobile1": "",

"mobile2": ""

}

],

"landlines": [

{

"landline1": "",

"landline2": ""

}

],

"emails": [

{

"email1": "",

"email2": ""

}

]

},"ownerEmployerOccupierDetails ":{"designation":"","name":"","address":{"premiseNoName": "",

"subLocalityColonyName": "",

"localityAreaSuburb": "",

"stateCode": "",

"districtCode": "",

"pinCode": "",

"country": ""

},"mobile":"","landline":"","email":""},

"contactPerson": [

{

"personName": "",

"roleDesignation": "",

"email": "",

"phone": ""

},

{

"personName": "",

"roledesignation": "",

"email": "",

"phone": ""

}

],

"actsApplicable": {

"acts": [

{

"actAcronym": "",

"coverageDate": "23-11-1999"

},

{

"actAcronym": "",

"coverageDate": "23-11-1999"

}

]

},

"identifiers": [

{

"identifierTypeAcronym": "",

"identifierValue": ""

},

{

"identifierTypeAcronym": "",

"identifierValue": ""

}

],"employerRegistrations":[{"employerRegistrationUnderAct":"","registrationNo":"","registrationDate":"dd-mm-yyyy","noOfWorkersAtTheTimeOfRegistration":0,"dateOfRenewalOfRegistration":"dd-mm-yyyy","noOfWorkersAtTheTimeOfRenewalOfRegistration":0,"licenseNo":"","licenseDate":"","dateOfRenewalOfLicense":"","noOfWorkersAtTheTimeOfRenewalOfLicense":0},{"employerRegistrationUnderAct":"","registrationNo":"","registrationDate":"dd-mm-yyyy","noOfWorkersAtTheTimeOfRegistration":0,"dateOfRenewalOfRegistration":"dd-mm-yyyy","noOfWorkersAtTheTimeOfRenewalOfRegistration":0,"licenseNo":"","licenseDate":"","dateOfRenewalOfLicense":"","noOfWorkersAtTheTimeOfRenewalOfLicense":0}],

"totalNumberOfWorkers": {

"wagesDrwanLessThenOrEqualTo24000":{"contractMaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0},

"contractFemaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0},

"regularMaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0},

"regularFemaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0}},"wagesDrwanMoreThen24000":{"contractMaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0},

"contractFemaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0},

"regularMaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0},

"regularFemaleWorkers": {"highlySkilled":0,"skilled":0,"semiSkilled":0,"unskilled":0}}},

"setup": "23-11-1999","establishmentWorkingStatus":"","ifFactoryRegistrationUnderSection ":"","ifFactoryPowerInHP":"","doEstablishmentNotifiedForAppointmentOfSafetyOfficer":"",

"isInvolvedInHazardousActivity": "",

"isInvolvedInDangerousActivity": "",

"isSeasonal": "",

"sectorAcronym": "",

"categoryAcronym": "",

"nicCode": ""

,"primaryBusinessActivity":"","natureOfWork":"","subCategoryOfNatureOfWork":"","breifDescription":""}

}

}

}

**Response & Error (It is same as in Version 1.0)**

## LIN Profile update

Service Name: update establishment representative.

URI: <http://api.shramsuvidha.gov.in/services/webservices/establishment/representative/update>

Method: PUT

**Input parameters**

{

"shramSuvidha":{"servicesForLINOnlyRegime":{"header":{"version":"2.1","serviceId":"UPDATEREPRESENTATIVE","serviceName":"Update Representative of LIN","method":"PUT","developedBy":"NIC Labour & Employment Division"},

"referenceParameters": {

"referenceNumber": "",

"referenceDate": "23-11-1999"

},"authenticationParameters": {

"userName": "",

"password": ""

},

"requestParameter": {

"lin":"125896368",

"establishmentRepresentative": {

"nameOfRepresentative": "","roleDesignation":"",

"emailId": "",

"mobile": "9868878798",

"address": {

"premiseNoName": "", "subLocalityColonyName": "", "localityAreaSuburb": "", "stateCode": "", "districtCode": "", "pinCode": "", "country": ""

}}}}}}

**Response Parameters**

**Success response:**

{

“successMessage”: “Establishment Representative Data Updated Successfully”,

“successCode”: 1xxx

}

**Error Response:**

{

"errorMesssage": "Lin must be 10 digits",

"errorCode": 5001,

"errorInfo": "it EITHER violates not-null constraint OR DUPLICATE\_KEY\_VALUE\_VIOLATES\_UNIQUE\_CONSTRAINT"

}

-------------------

{

"errorMesssage": "Please provide district code",

"errorCode": 5001,

"errorInfo": "it EITHER violates not-null constraint OR DUPLICATE\_KEY\_VALUE\_VIOLATES\_UNIQUE\_CONSTRAINT"

}

## Lin Profile Filter By District Id Or State Id

Following is the sample data for ‘**Lin Profile Filter by District id or state id**’ web service:

**Example Request**

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET (LP11-GET)** | http://api.shramsuvidha.gov.in/services/webservice/lin/profile/filter/{code} |

|  |
| --- |
|  |

**Request**

|  |
| --- |
| http://api.shramsuvidha.gov.in/services/webservice/lin/profile/filter/265 |

**Response**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"Version": "2.1",

"serviceId": "LINPROFILEBYSTATEDISTCODE",

"serviceName": "LIN Profile Filter by State/District Code",

"method": "GET",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"codeProvided": "265"

,"districtCode":"265","districtName":"Dimapur","StateCode":"13","StateName":"Nagaland"},

"message": {

"MessageCode": "1001",

"MessageText": "Success"

},

"linProfile": {

"lin":"X-XXXX-XXXX-X","EstablishmentName": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"contacts": [

{

"contactType": "",

"conatactValue": ""

},

{

"contactType": "",

"conatactValue": ""

}

],

"contactPersons": [

{

"name": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"roleRepresentation": ""

},

{

"name": "",

"address": {

"premiseNoName": "",

"subLocalityColony": "",

"localityAreaSuburb": "",

"townCity": "",

"district": "",

"pinCode": "",

"state": "",

"country": ""

},

"roleRepresentation": ""

}

],

"setupDate": "23-11-1999",

"sector": "",

"category": "",

"involvedInHazardousActivity": "",

"involvedInDangerousActivity": "",

"seasonal": "",

"totalNumberOfEmployeesAtTheTimeOfLINAllotment": "",

"natureOfActivity": "",

"briefDescription": "",

"identifiers": [

{

"identifierType": "",

"identifierValue": ""

},

{

"identifierType": "",

"identifierValue": ""

}

],

"entityRelationship": [

{

"relatedEntityLIN": "",

"entityName": "",

"relationship": ""

},

{

"relatedEntityLIN": "",

"entityName": "",

"relationship": ""

}

]

}

}

}

}

**Error Message**

{

"shramSuvidha": {

"servicesForLINOnlyRegime": {

"header": {

"version": "2.1",

"serviceId": "LINPROFILEBYSTATEDISTCODE",

"serviceName": "LIN Profile Filter by State/District Code",

"method": "GET",

"developedBy": "NIC Labour & Employment Division"

},

"requestParameter": {

"codeProvided": "100"

,"districtCode":"Null","districtName":"Null","stateCode":"Null","stateName":"Null"},

"errorMessage": {

"messageCode": "3001",

"messageText": "Success"

}}

}

}

# Annexure

## Message Code

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Code** | **Message (s)** |
| 1 | **1xxx** | Success |
| 2 | **2xxx** | Desired information not available |
| 3 | **3xxx** | No information available about the requested parameter |
| 4 | **4xxx** | Invalid User-id/Password |
| 5 | **5xxx** | Multiple messages are sent to identify the missing mandatory information like Office Acronym not available, Identifier Acronym not given or user-id itself is not available |
| 7 | **7xxx** | Multiple messages indicating invalid values of the parameters/arguments like wrong Office Acronym, wrong Identifier Acronym |
| 8 | **8xxx** | Invalid Schema |
|  | **8001** | Schema out of order |
|  | **8002** | Field name mismatch |
|  | **8003** | Extra field received |
|  | **8004** | Wrong field name |
| 9 | **9xxx** | System error |

## Recommendations

1. If client is using SSL then Shram Suvidha Services are not built for Ajax call. Client might get below error:

**Error:** Mixed Content: The page at 'https: //shramsuvidha.gov.in/ip' was loaded over HTTPS, but requested an insecure XMLHttpRequest endpoint 'http: //clientsite.com'. This content should also be served over HTTPS.

**Solution:** Don't use Ajax, write code on backend application.

1. If client is on NIC network then can directly hit Shram Suvidha IP's.

# Privacy Policy

Information about web services in this document is for consumption by various authorities, it is assumed that privacy of profile information of individuals and establishments shared via these web services is maintained and profile’s information should not get used for unintended purpose. Please use web services in a way that the privacy of individuals and establishments is maintained.

Shram Suvidha API Portal does not automatically capture any specific personal information from any source, (like name, phone number or e-mail address), that allows us to identify you individually or as establishment.

This portal records your requests and response for statistical purposes, We make no attempt to link these data with the identity of individuals or establishments with any external means. It is also expected that Service Consumers should also not share any data shared via API communication to any external means or third party (public/private). Any information provided to this website will be protected from loss, misuse, unauthorized access or disclosure, alteration, or destruction and reasonable security measures such as administrative, technical, operational and physical controls have been implemented to ensure the security of personal information, if collected.

If at any time you believe the principles referred to in this privacy statement have not been followed, or have any other comments on these principles, please notify this at [tech-shramsuvidha@nic.in](mailto:tech-shramsuvidha@nic.in)