

Contents

Add Suburbs API Specification	2
Request Parameters	2
Response Parameters:	2
Validation Rules	3
Sample Request	3
Sample Success Response	4
Sample Error Response	4
Search Suburbs API Specification	5
Validation Rules	5
Request Parameters	5
Response Parameters:	6
Sample Request	7
Sample Success Response	7
Sample Error Response	7
Error codes	8

Add Suburbs API Specification

This API adds the suburb details provided in the request to DB.

Method	URL
POST	http://<serverip>:<port>/suburbs/add

Request Parameters

Name	Type	Length	Description	Required/Optional/Conditional
AddSuburbRequest				
suburbDetails	List<Suburb>		List of suburbs to be added	Required

Name	Type	Length	Description	Required/Optional/Conditional
Suburb				
name	String		Name of the Suburb	Required
postCode	String		Postcode of the Suburb	Required

Response Parameters:

Name	Type	Length	Description	Required/Optional/Conditional
ServiceResponse				
responseContext	Object		Contains the response code and error information if any.	Required

Name	Type	Length	Description	Required/Optional/Conditional
ResponseContext				
responseCode	String		Response Code of the request "000" – Success "001" - invalid parameter "999" – Internal System error	Required
responseDescription	String		Response Description "Success" or failure description if case of failure	Required

Validation Rules

- If any suburb name already exists in DB the record will not be saved. No error is thrown.
- Suburb name and postcode must be mandatorily provided. The entry which does not have the name or postcode details will not be saved. The valid records will be saved. No error is thrown.

Sample Request

```
{
  "suburbDetails" : [
    {
      "name" : "Perth",
      "postCode" : "6001"
    },
    {
      "name" : "East Perth",
      "postCode" : "6004"
    }
  ]
}
```

Sample Success Response

```
{
  "responseContext" : {
    "responseCode" : "000",
    "responseDescription" : "Success"
  }
}
```

Sample Error Response

```
{
  "responseContext" : {
    "responseCode" : "999",
    "responseDescription" : "Internal Server Error"
  }
}
```

Search Suburbs API Specification

This API returns a list of suburb names for the requested postcode range. It also returns the total number of characters of the suburb names.

Method	URL
POST	http://<serverip>:<port>/suburbs/search

Validation Rules

- SearchCriteria cannot be empty or null.
- Start postcode and end postcode must be provided.
- Start postcode should be less than the end postcode.

Request Parameters

Name	Type	Length	Description	Required/Optional/Conditional
serviceRequestBody				
searchCriteria	Object		Search Criteria	Required

Name	Type	Length	Description	Required/Optional/Conditional
SearchCriteria				
startPostCode	String		Start of Postcode range to search	Required
endPostCode	String		End of Postcode range to search	Required

Response Parameters:

Name	Type	Length	Description	Required/Optional/Conditional
SearchSuburbsResponse				
responseContext	Object		Contains the response code and error information if any.	Required
suburbDetails	List<String>		Names of the suburbs within the provided range of postcodes. Sorted in alphabetical order.	Required
totalCharactersCount	Integer		Total number of characters of all the resultant suburb names	Required

Name	Type	Length	Description	Required/Optional/Conditional
ResponseContext				
responseCode	String		Response Code of the request "000" – Success "001" - Invalid Request "002" - Missing Parameter in Request "003" - No results found "999" – Internal System error	Required
responseDescription	String		Response Description "Success" or failure description if case of failure	Required

Sample Request

```
{
  "searchCriteria" : {
    "endCode" : "6001",
    "startCode" : "6003"
  }
}
```

Sample Success Response

```
{
  "responseContext" : {
    "responseCode" : "000",
    "responseDescription" : "Success"
  },
  "suburbDetails" : [ "Suburb1", "Suburb2", "Suburb3" ],
  "totalCharactersCount" : 21
}
```

Sample Error Response

```
{
  "responseContext" : {
    "responseCode" : "001",
    "responseDescription" : "Invalid Request"
  }
}
```

Error codes

- 1) 001 – Invalid Request
- 2) 002 - Missing Parameter in Request
- 3) 003 - No results found
- 4) 999 – Internal Server Error