**Locations**

URIs relative to <http://localhost:8080>

Location Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /location

Request Body:

{ “name” : string}

Will Return 400 Error if location is already in Database

**Get Method**

HTTP Request GET /location

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /location

Request Body:

{“name” : string}

Will return 400 error if location does not exist in Database

**Languages**

URIs relative to <http://localhost:8080>

Language Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /language

Request Body:

{ “name” : string}

Will Return 400 Error if language is already in Database

**Get Method**

HTTP Request GET /language

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /language

Request Body:

{“name” : string}

Will return 400 error if language does not exist in Database

**Race**

URIs relative to <http://localhost:8080>

Race Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /race

Request Body:

{ “name” : string}

Will Return 400 Error if race is already in Database

**Get Method**

HTTP Request GET /race

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /race

Request Body:

{“name” : string}

Will return 400 error if race does not exist in Database

**Ethnicity**

URIs relative to <http://localhost:8080>

Ethnicity Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /ethnicity

Request Body:

{ “name” : string}

Will Return 400 Error if ethnicity is already in Database

**Get Method**

HTTP Request GET /ethnicity

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /ethnicity

Request Body:

{“name” : string}

Will return 400 error if ethnicity does not exist in Database

**Smoking History**

URIs relative to <http://localhost:8080>

Smoking history Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /smoking-history

Request Body:

{ “name” : string}

Will Return 400 Error if smoking-history is already in Database

**Get Method**

HTTP Request GET /smoking-history

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /smoking-history

Request Body:

{“name” : string}

Will return 400 error if smoking-history does not exist in Database

**Appetite**

URIs relative to <http://localhost:8080>

Appetite Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /appetite

Request Body:

{ “name” : string}

Will Return 400 Error if appetite is already in Database

**Get Method**

HTTP Request GET /appetite

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /appetite

Request Body:

{“name” : string}

Will return 400 error if appetite does not exist in Database

**Supplements**

URIs relative to <http://localhost:8080>

Supplement Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /supplement

Request Body:

{ “name” : string}

Will Return 400 Error if supplement is already in Database

**Get Method**

HTTP Request GET /supplement

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /supplement

Request Body:

{“name” : string}

Will return 400 error if supplement does not exist in Database

**Musculoskeletal**

URIs relative to <http://localhost:8080>

Musculoskeletal Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /musuloskeletal

Request Body:

{ “name” : string}

Will Return 400 Error if musuloskeletal is already in Database

**Get Method**

HTTP Request GET /musuloskeletal

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /musuloskeletal

Request Body:

{“name” : string}

Will return 400 error if musuloskeletal does not exist in Database

**Bed**

URIs relative to <http://localhost:8080>

Bed Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /bed

Request Body:

{ “name” : string}

Will Return 400 Error if bed is already in Database

**Get Method**

HTTP Request GET /bed

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /bed

Request Body:

{“name” : string}

Will return 400 error if bed does not exist in Database

**Chair**

URIs relative to <http://localhost:8080>

Chair Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /chair

Request Body:

{ “name” : string}

Will Return 400 Error if chair is already in Database

**Get Method**

HTTP Request GET /chair

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /chair

Request Body:

{“name” : string}

Will return 400 error if chair does not exist in Database

**Feet**

URIs relative to <http://localhost:8080>

Feet Representation

{

“name”: *string*

}

**Create Method**

HTTP Request: POST /feet

Request Body:

{ “name” : string}

Will Return 400 Error if feet is already in Database

**Get Method**

HTTP Request GET /feet

Response:

[

{

“name”:string

},

{

“name”:string

},

…

{

“name”:string

}

]

**Delete Method**

HTTP Request: DELETE /feet

Request Body:

{“name” : string}

Will return 400 error if feet does not exist in Database