RPM API Contract

**Enterprise**

* Create Enterprise……………………………………………………………………3
* Update Enterprise by ID……………………………………………………………………3
* Get all Enterprises ……………………………………………………………………4
* Get Enterprise by ID ……………………………………………………………………3
* Delete Enterprise by ID ……………………………………………………………………5

**Practice**

* Create Practice by enterpriseId……………………………………………………………….6
* Update Practice by ID…………………………………………………………………….6
* Get All Practice by enterpriseId …………………………………………………………………….7
* Get Practice by ID …………………………………………………………………….7
* Delete Practice by ID…………………………………………………………………….8-9

**Patient**

* Create Patient....……………………………………………………………….11
* Update Patient by ID…………………………………………………………………….11-12
* Update Patient Image by ID............................................................13
* Update Patient Expired by ID ........................................................13
* Get All Patient by PracticeId …………………………………………………………………….15-16
* Delete Patient by ID…………………………………………………………………….16

**PatientObs**

* Create PatientObs……………………………………………………………….17
* Get PatientObs by PatientId …………………………………………………………………….19
* Delete PatientObs By ID…………………………………………………………………….19

**External System**

* Create External System …………………………………………………………………...22
* Update External System by ID………………………………………………………............……………23
* Get External System by ID…………………………………………………………………………23
* Get All External System ………………………………………………………23
* Delete External System by ID ………………………………………………………24

**External System Param**

* Create External System Param ………………………………………………………….25
* Update Extrenal System Param by ID…………………………………………………………………………27
* Get External System by ID...............................................................................26
* Get All External System Param ……………………………………………………….26
* Delete External System Param by ID………………………………………………………………………….28

**External Patient Map**

* + Create External Patient Map ..............................................................................29
  + Get External Patient Map by External ID................................................29

**External Obsterm Map**

* Create External Obsterm Map........................................................................32
* Get External Obsterm Map by External ObsertermID.......................................................32
* Update External Obsterm Map by ID..............................................................34
* Delete External Obsterm map by ID ......................................................................35

**Obsterm**

* Get Obsterm by ID....................................................36
* Get All Obsterm.........................................................................................36

**Code System**

* Get CodeSystem by Code.......................................................................................37
* Get All CodeSysetm...............................................................................38

**Unit Of Measurement**

* Get UOM by Code........................................................................................38
* Get All UOM.........................................................................39

**Obsterm Range**

* Get Obsterm Range by ID.....................................................................................40

**Enterprise**

* **Create Enterprise**  
    
  **URL: http(s)://<hostname>:<port>/v1/enterprises**  
    
  **Http Method: POST**  
    
  **Request Payload:**  
    
  {  
    
  “name”: ”Apollo”  
    
  }  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**  
    
  {  
    
  “enterpriseId” : 10000001  
    
  }
* **Update Enterprise by ID**   
    
  **URL:** **http(s)://<hostname>:<port>/v1/ enterprises/{enterpriseId}**   
    
  **Http Method: PUT**   
    
  **Request Payload:**    
    
  {   
    
  “name” : ” ”   
    
  }   
    
  **Response HTTP Status: 200 OK**  
    
  **Response :**   
    
  {  
    
  “message” : “Enterprise updated successfully.”  
    
  }  
    
  **If error -**  
  {  
  "error":{"code": 404,  
   "message": "The requested enterprise ID could not be found"  
   }  
    
  }
* **Get Enterprise By ID**  
    
  **URL**: **http(s)://<hostname>:<port>/v1/ enterprises/{enterpriseId}**   
    
  **Http Method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  {   
    
  “name” : “ ”  
    
  }   
    
  **If error -**  
    
  {  
  "error": {  
  "code": 404,  
  "message": "The requested enterprise ID could not be found"  
  }  
  }
* **Get All Enterprises**   
    
  **URL**: **http(s)://<hostname>:<port>/v1/ enterprises**  
    
  **Http Method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  [  
    
  {   
    
  “enterpriseId” : 10000001  
    
  “name” : ” ”  
    
  },  
    
  {   
    
  “enterpriseId” : 10000002  
    
  “name” : ” ”  
    
  }  
    
  ]
* **Delete Enterprise by ID**  
    
  **URL**: **http(s)://<hostname>:<port>/v1/ enterprises/{enterpriseId}**  
    
  **Http Method: DELETE**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  {  
    
  “message” : “Enterprise deleted successfully.”  
    
  }  
    
  **If error -**   
    
  {  
  "error": {  
  "code": 404,  
  "message": "The requested enterprise ID could not be found"  
  }  
  }  
    
     
    
   **Practice**
* **Create Practice by EnterpriseId**  
    
  **URL: http(s)://<hostname>:<port>/v1/enterprises/{enterpriseId}/practices**  
    
  **Http Method: POST**  
    
  **Request Payload**:  
    
  {

"name": “”  
  
 }  
  
  
 **Response HTTP Status: 200 OK**  
  
 **Response:**  
  
 {  
  
 “practiceId” : 100001  
  
 }

* **Update Practice by ID**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/enterprises/{enterpriseId}/practices/{practiceId}**  
    
  **Http method: PUT**  
    
  **Request Payload:**  
    
   {

“name": ""

}

* **Get Practice by ID**  
    
  **URL: http(s)://<hostname>:<port>/v1/practices/{practiceId}**  
    
  **Http Method: GET**  
    
  **Response Http status: 200 OK**  
    
  **Response :**  
    
  {

“enterpriseId”:100001,  
  
 "name": ""   
  
}  
  
 **If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested practice ID could not be found"  
}  
}

* **Get All Practices by EnterpriseId**  
    
  **URL: http(s)://<hostname>:<port>/v1/enterprises/{enterpriseId}/practices**  
    
  **Http Method: GET**  
    
  **Response Http status: 200 OK**

**Response :**  
  
[  
  
{   
  
 “practiceId” : 10000001,  
  
 "name": ""   
  
},  
{   
  
 “practiceId” : 10000002,  
  
 "name": ""   
  
}  
]  
  
  
**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested enterprise ID could not be found"  
}  
}

* **Delete Practice by ID**
* **URL**: **http(s)://<hostname>:<port>/v1/ practices/{practiceId}**  
    
  **Http Method: DELETE**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  {  
    
  “message” : “Practice deleted successfully.”  
    
  }  
    
  **If error -**   
    
  {  
  "error": {  
  "code": 404,  
  "message": "The requested practice ID could not be found"  
  }  
  }

**Patient**

* **Create patient**   
    
  **URL:** **http(s)://<hostname>:<port>/v1/practices/{practiceId}/patients**  
    
  **Http method: POST**  
    
  **Request Payload:**  
    
  {  
   "firstName": "string",  
    
   "middleName": "string",  
    
   "lastName": "string",  
    
   "preferredName": "string",  
    
   "birthDate": ,  
    
   "gender": "string",  
    
  "birthSex": "string",  
    
   "maritalStatus": "string",  
    
   "emailId": "string",  
    
   "phoneNumber":

}  
  
**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“patientId”:10001  
  
}

* **Update patient By ID**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/patients/{patientId}**  
    
  **Http method: PUT**
* **Request Payload:**  
    
  {   
   "firstName": "string",  
    
   "middleName": "string",  
    
   "lastName": "string",  
    
   "preferredName": "string",  
    
   "birthDate": ,  
    
   "gender": "string",  
    
   "birthSex": "string",  
    
   "maritalStatus": "string",  
    
   "emailId": "string",  
    
   "phoneNumber": 8021231223  
    
  }

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested patient ID could not be found"  
}  
}

* **Update Patient Image by ID**  
    
  **URL: http(s)://<hostname>:<port>/v1/patients/{patientId}/patient\_photo**  
    
  **Http method: PATCH**  
    
  **Request Payload**:  
    
  {  
    
  “photoUrl” : “”  
    
  }  
    
  **Response Http status: 200 OK**  
    
  **Response :**  
    
  {  
    
  “message” : “Patient Image updated successfully.”  
    
  }  
    
  **If error -**  
    
  {  
  "error": {  
  "code": 404,  
  "message": "The requested patient ID could not be found"  
  }  
  }
* **Update Patient Expired by ID**  
    
  **URL: http(s)://<hostname>:<port>/v1/patients/{patientId}**  
    
  **Http method: PATCH**  
    
  **Request Payload**:  
    
  {  
    
   "expired":true,  
    
   "expiredDate": 2023-03-20  
    
  }  
    
  **Response Http status: 200 OK**  
    
  **Response :**  
    
  {  
    
  “message” : “Patient Image updated successfully.”  
    
  }  
    
  **If error -**  
    
  {  
  "error": {  
  "code": 404,  
  "message": "The requested patient ID could not be found"  
  }  
  }
* **Get Patient by ID**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/patients/{patientId}**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  {  
   "firstName": "string",  
    
   "middleName": "string",  
    
   "lastName": "string",

“photoUrl” : “”  
  
 "preferredName": "string",  
  
 "birthDate": ,  
  
 "gender": "string",  
  
 "birthSex": "string",  
  
 "maritalStatus": "string",  
  
 "emailId": "string",

"phoneNumber": integer,  
  
 "expired":true,  
  
 "expiredDate”: 20230220  
  
}  
  
**If error -**  
  
{  
"error": {  
"code": 404,  
"message": "The requested patient ID could not be found"  
}  
}

* **Get All Patient by practiceId**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/practices/{practiceId}/patients**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  [  
    
  {  
    
   “patientId”:100002,  
    
   “photoUrl”:” ”,  
    
   "firstName": "string",  
    
   "middleName": "string",  
    
   "lastName": "string",  
    
   "preferredName": "string",  
    
   "birthDate": ,  
    
   "gender": "string",  
    
   "birthSex": "string",  
    
   "maritalStatus": "string",  
    
   "emailId": "string",  
    
   "phoneNumber": integer,  
    
   "expired":boolean,  
    
   "expiredDate":20230220  
     
  }

]

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested practice ID could not be found"  
}  
}

* **Delete Patient by ID**

**URL: http(s)://<hostname>:<port>/v1/patients/{patientId}**  
  
**Http method: DELETE**  
  
**Response HTTP Status: 200 OK**  
  
**Response :**

{  
 “message”: ”patient deleted successfully”  
 }

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested patient ID could not be found"  
}  
}

**PatientObs**

* **Create PatientObs**

**URL:** **http(s)://<hostname>:<port>/v1/patientObs**  
  
**Http method: POST**  
  
**Request Payload:**  
  
{  
  
“patientId”: 100001,

“obstermId” : 100001,

“value” : 120,

“uomCode” : “”,

“effectiveDateTime”:  
  
}  
  
**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“patientObsId”:1000001  
  
}

* **Get PatientObs by patientId**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/patients/{patientId}/patientObs**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**   
    
  [

{

“obstermId” : 100001,

“value” : 120,

“uomCode” : “”,

“effectiveDateTime”:

}

]

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested patient ID could not be found"  
}  
}

**PatientObs Details**

* **Get All PatientObs Details**   
    
  **URL:** **http(s)://<hostname>:<port>/v1/{patientObsId}/patientObsDetails**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

[

{

“patientObsDetailsId” : 10001

“patientObsId” : 10001,

“obstermId” : 10003,

“obsTypeId” : 3

}

]

**External System**

* **Create External System**

**URL:** **http(s)://<hostname>:<port>/v1/externalSystems**  
  
**Http method: POST**  
  
**Request Payload:**  
  
{

“name”: ”Medelinked”

}

**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“externalSystemId”:1000001  
  
}

* **Get External System by ID**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/externalSystems/{externalSystemId}**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

{

“name”: ”Medelinked”

}

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested External System by ID could not be found"  
}  
}

* **Get All External System**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/externalSystems**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

[

{

“externalSystemId”:1000001,

“name”: ”Medelinked”

}

]

* **Update External System By ID**  
    
  **URL: http(s)://<hostname>:<port>/v1/ externalSystems/{externalSystemId}**  
    
  **Http method: PUT**  
    
  **Request Payload**:  
    
  {  
    
   "name”:”validic”  
    
  }  
    
  **Response Http status: 200 OK**  
    
  **Response :**  
    
  {  
    
  “message” : “External System updated successfully.”  
    
  }  
    
  **If error -**  
    
  {  
  "error": {  
  "code": 404,  
  "message": "The requested External System ID could not be found"  
  }  
  }

**External System Param**

* **Create External System Param**

**URL:** **http(s)://<hostname<port>/v1/externalSystems/{externalSystemId}/externalSystemParams**  
  
**Http method: POST**  
  
**Request Payload:**  
  
{

"key”:””,

“value”:””

}

**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“externalSystemParamId”:1000001  
  
}

* **Get External System Param by ID**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/externalSystems/{externalSystemId}/externalSystemParams/{externalSystemParamId}**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

{

“externalSystemParamId”:10001,

"key”:””,

“value”:””

}

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested External System Param by ID could not be found"  
}  
}

* **Get All External System Param**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/externalSystems/{externalSystemId}/externalSystemParams**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

[

{

“externalSystemParamId”:100001

“externalSystemId”:10001,

"key”:””,

“value”:””

}

]

* **Update External System Param By ID**  
    
  **URL: http(s)://<hostname>:<port>/v1/ externalSystems/{externalSystemId} /externalSystemParam/{externalSystemParamId}**  
    
  **Http method: PUT**  
    
  **Request Payload**:  
    
  {  
    
   "key”:””,

“value”:””

}  
  
**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“message” : “External System updated successfully.”  
  
}  
  
**If error -**  
  
{  
"error": {  
"code": 404,  
"message": "The requested External System Param ID could not be found"  
}  
}

**External Patient Map**

* **Create External Patient Map**

**URL:http(s)://<hostname>:port>/v1/externalSystems/{externalSystemId}/externalPatientMappings**  
  
**Http method: POST**  
  
**Request Payload:**  
  
{

“patientId” : “p1”,

“externalPatientId” : “u1”

}

**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“externalPatientMap”:1000001  
  
}

* **Get External Patient Map by externalPatientId**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/externalPatientMappings/{externalPatientId}**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

{

“externalPatientMapId”:10001,

“externalSystemId”:10001,

“patientId” : “p1”

}

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested External Patient Map ID could not be found"  
}  
}

**External Obsterm Map**

* **Create External Obsterm Map**

**URL:http(s)://<hostname>:port>/v1/externalSystems/{externalSystemId}externalObstermMap**  
  
**Http method: POST**  
  
**Request Payload:**  
  
{

“obsterm” : “obsterm1”,

“externalObstermId” : “OBS1”

}

**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“externalObstermMap”:1000001  
  
}

* **Get External Obsterm Map by External Obsterm Id**  
    
  **URL:** **http(s)://<hostname>:<port>/v1/externalObstermMap/{externalObstermId}**  
    
  **Http method: GET**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response:**

{

“externalObstermMapId”: 10001

“externalSystemId”:10001,

“obsterm” : “obsterm1”,

}

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested External Obsterm Map ID could not be found"  
}  
}

* + - **Update External Obsterm Map By ID**  
        
      **URL: http(s)://<hostname>:<port>/v1/ externalSystems/{externalSystemId} /externalObstermMap/{externalObstermMapId}**  
        
      **Http method: PUT**  
        
      **Request Payload**:  
        
      {  
       “obsterm” : “obsterm1”,

“externalObstermId” : “OBS1”

}  
  
**Response Http status: 200 OK**  
  
**Response :**  
  
{  
  
“message” : “External Obsterm Map updated successfully.”  
  
}  
  
**If error -**  
  
{  
"error": {  
"code": 404,  
"message": "The requested External Obsterm Map ID could not be found"  
}  
}

* **Delete External Patient Map By ID**  
    
  **URL: http(s)://<hostname>:<port>/v1/externalObstermMap/{externalObstermMapId}**  
    
  **Http method: DELETE**  
    
  **Response HTTP Status: 200 OK**  
    
  **Response :**

{

“message”: ” External Obsterm Map deleted successfully”

}

**If error -**  
  
{  
"error": {  
"code": 404,  
"message": "The requested External Obsterm Map ID could not be found"  
}  
}

**Obsterm**

* **Get Obsterm by ID**

**URL: http(s)://<hostname>:<port>/v1/obsTerms/{obsTermId}**

**Http method: GET**

**Response HTTP Status: 200 OK**

**Response:**

{

“name”:”Heart Rate”,

“code”:” heart-rate”,

“obsTypeId”:1,

“obsCategoryId”:1

}

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested Obsterm ID could not be found"  
}  
}

* **Get All Obsterm**

**URL: http(s)://<hostname>:<port>/v1/obsTerms**

**Http method: GET**

**Response HTTP Status: 200 OK**

**Response:**

[

{

“obstermId”:10001,

“name”:”Heart Rate”,

“code”:”heart-rate”,

“obsTypeId”:1,

“obsCategoryId”:1

}

]

**Unit Of Measurement**

* **Get UOM by Code**

**URL: http(s)://<hostname>:<port>/v1/uoms/{code}**

**Http method:GET**

**Response HTTP Status: 200 OK**

**Response:**

{

“displayName”:”percent”,

“ucmcd”:”%”

}

**If error -**   
  
{  
"error": {  
"code": 404,  
"message": "The requested Uom Code could not be found"  
}  
}

* **Get All UOM**

**URL: http(s)://<hostname>:<port>/v1/uoms**

**Http method: GET**

**Response HTTP Status: 200 OK**

**Response:**

[

{

“code”:”%”,

“displayName”:”percent”,

“ucmcd”:”%”

}

]