**DAVICOM Semiconductor, Inc.**

Web API 規格書

Web API格式使用說明

***YiCheng-Liao***

***Micron\_WebAPI規格書\_N01\_zh\_TW***

***February 15, 2023***

History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本** | **頁數** | **說明** | **修改者** | **備註** |
| N01 |  | 初搞 | YiCheng | 2023-02-06 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目 錄

[History ii](#_Toc126754553)

[一、 資料庫定義 4](#_Toc126754554)

[1. ERP關聯資料表 4](#_Toc126754555)

[二、 使用者資料表 5](#_Toc126754556)

[1. 建立使用者資料 5](#_Toc126754557)

[2. 查詢使用者 6](#_Toc126754558)

[3. 查詢使用者ID 7](#_Toc126754559)

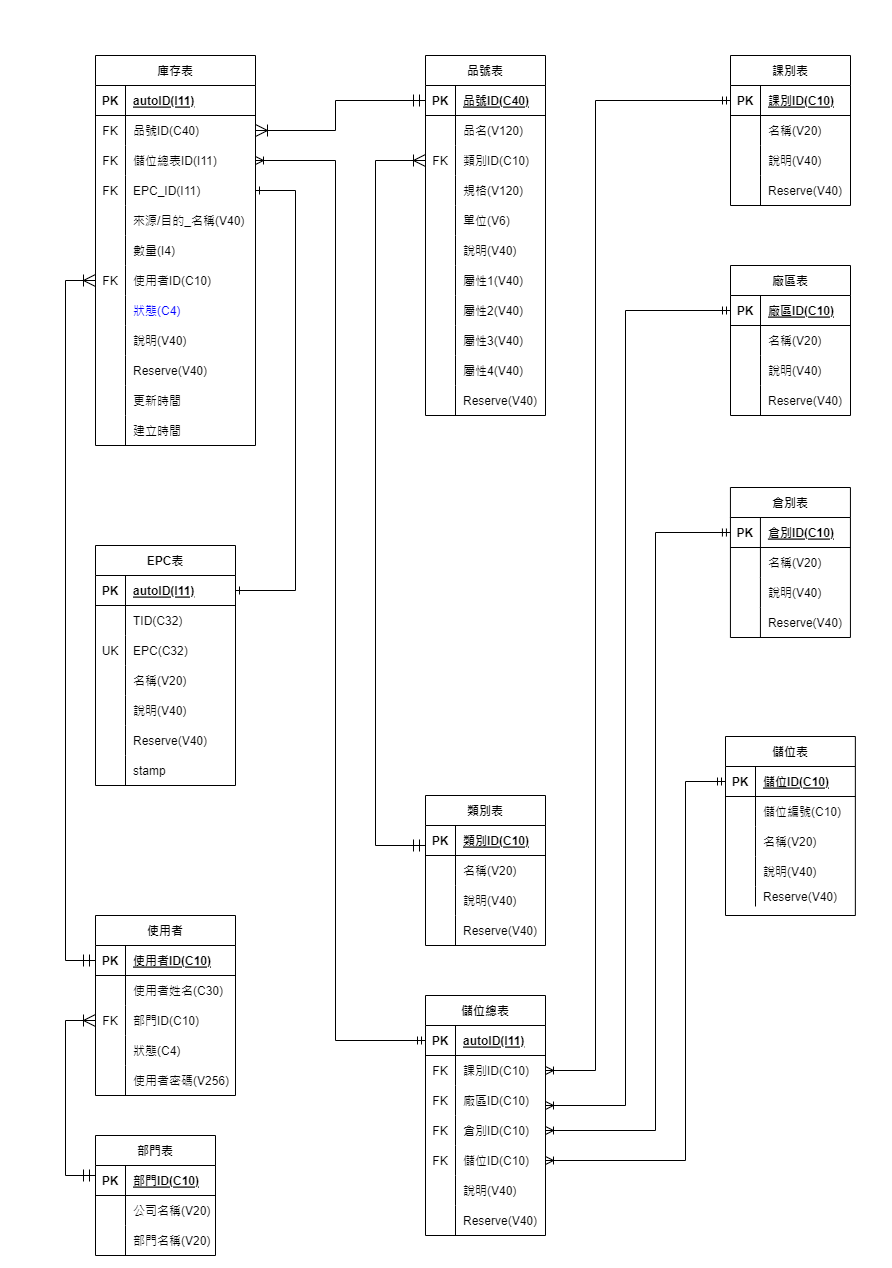
[4. 更新使用者 8](#_Toc126754560)

[5. 刪除使用者 9](#_Toc126754561)

[三、 品號資料表 10](#_Toc126754562)

# 資料庫定義

## ERP關聯資料表



原始檔案: 美光流程資料庫V1.5.drawioo

# 名稱說明

|  |  |
| --- | --- |
| Request | |
|  |  |
| Response "service": 0=sys, 1=web, 2=database, 3=other …. ,server處理回應代號  "code": 依server系統訊息碼回應  "msg": 文字簡要說明  "data":[{json data}], Array以 JSON詳細資料列表  "service": 1 && "code": 200 為正確回應，其它為異常代號。 | |
|  | |
|  | |
|  | |

# 使用者資料表

|  |  |
| --- | --- |
| 建立使用者資料 | |
| 功能說明 | **建立單筆使用者資料** |
| url: <http://192.168.248.34:3000/catalog/user>  Method: POST | |
| Body:  {  "user\_id": "user006",  "name": "user0012",  "deptid": "dept333",  "state": "0",  "password": "123456"  } | |
| res.status(200); //http status  Response data:  {  "service": 1,  "code": 200,  "msg": "Flds\_user creat successfully user\_id: user006",  "data": []  }  ERROR:  res.status(200);  Response data:  {  "service": 2,  "code": 1062,  "msg": "ER\_DUP\_ENTRY",  "data": [    {  "msg": "Duplicate entry 'user006' for key 'PRIMARY'"  }  ]  } | |
|  | |

|  |  |
| --- | --- |
| 查詢使用者 | |
| 功能說明 | **查詢所有使用者** |
| |  | | --- | | url: <http://192.168.248.34:3000/catalog/user>  Method: GET | | Body:  無 | |  |   res.status(200); //http status  Response: 多筆JSON資料  {  "service": 1,  "code": 200,  "msg": "flds\_user findAll successfully....",  "data": [    {  "user\_id": "user001",  "name": "user0012",  "deptid": "dept001",  "state": "12",  "password": "123456",  "createtime": "2023-02-08 23:00:06",  "updatetime": "2023-02-08 23:00:06"  },    {  "user\_id": "user002",  "name": "user0012",  "deptid": "dept001",  "state": "",  "password": "123456",  "createtime": "2023-01-02 22:26:59",  "updatetime": "2023-02-08 18:58:38"  }  ]  } | |
|  | |

|  |  |
| --- | --- |
| 查詢使用者ID | |
| 功能說明 | **以ID查詢使用者** |
| url: <http://192.168.248.34:3000/catalog/user/user001>  Method: GET  req.params = user001 | |
| Body:  無 | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "Flds\_user findOne: user001 successfully.",  "data": [    {  "user\_id": "user001",  "name": "user0012",  "deptid": "dept001",  "state": "12",  "password": "123456",  "createtime": "2023-02-08 23:00:06",  "updatetime": "2023-02-08 23:00:06"  }  ]  }  res.status(200);  Response:查詢無資料  {  "service": 2,  "code": 404,  "msg": "Not found user\_id: user001",  "data": []  } | |
|  | |

|  |  |
| --- | --- |
| 更新使用者 | |
| 功能說明 | **以ID更新使用者資料** |
| url: <http://192.168.248.34:3000/catalog/user/user001>  Method: PUT  req.params = user001 | |
| Body:  {  "users\_id": "user001",  "name": "user0012",  "deptid": "dept001",  "state": "0",  "password": "654321"  } | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "Flds\_user updateById user\_id: user001 successfully.",  "data": []  }  res.status(200);  Response:無資料  {  "service": 2,  "code": 404,  "msg": "Not found id: user001",  "data": []  } | |
|  | |

|  |  |
| --- | --- |
| 刪除使用者 | |
| 功能說明 | **以ID更新使用者資料** |
| url: <http://192.168.248.34:3000/catalog/user/user006>  Method: DELETE | |
| Body:  無 | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "deleteOne successfully user\_id: user006",  "data": []  }  res.status(200);  Response:無資料  {  "service": 2,  "code": 404,  "msg": "Delete not found id: user006",  "data": [],  } | |
|  | |

|  |  |
| --- | --- |
| 查詢參數條件輸入 | |
| 功能說明 | **以參數LIKE查詢使用者資料**  Query: user\_id, deptid, LIMIT, OFFSET |
| url: <http://192.168.248.34:3000/catalog/user/findsearch?limit=9&offset=2>  Method: GET  req.params = findsearch  req.query = ?limit=9&offset=2 | |
| Body:  無 | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "Flds\_user findOne: user001 successfully.",  "data": [    {  "user\_id": "user001",  "name": "user0012",  "deptid": "dept001",  "state": "12",  "password": "123456",  "createtime": "2023-02-08 23:00:06",  "updatetime": "2023-02-08 23:00:06"  }  ]  }  res.status(200);  Response:查詢無資料  {  "service": 2,  "code": 404,  "msg": "Not found user\_id: user001",  "data": []  } | |
|  | |

# 品號資料表

|  |  |
| --- | --- |
| 建立品號資料 | |
| 功能說明 | **建立單筆品號資料** |
| url: <http://192.168.248.34:3000/catalog/product/>  Method: POST | |
| Body:  {  "product\_no": "220309208600000046",  "product\_name": "product\_name0000",  "classift": "class\_0001",  "specification": "specification\_0000",  "unit": "支",  "remake": "Test0000",  "attribute1": "",  "attribute2": "",  "attribute3": "",  "attribute4": "",  "reserve": ""  } | |
| res.status(200); //http status  Response data:  {  "service": 1,  "code": 200,  "msg": "product creat successfully product\_no: 220309208600000046",  "data": []  }  ERROR:  res.status(200);  Response data:  {  "service": 2,  "code": 1062,  "msg": "ER\_DUP\_ENTRY",  "data": [    {  "msg": "Duplicate entry '220309208600000046' for key 'PRIMARY'"  }  ],  } | |
|  | |

|  |  |
| --- | --- |
| 查詢所有品號 | |
| 功能說明 | **查詢所有品號資料** |
| |  | | --- | | url: <http://192.168.248.34:3000/catalog/product/>  Method: GET | | Body:  無 | |  |   res.status(200); //http status  Response: 多筆JSON資料  {  "service": 1,  "code": 200,  "msg": "product findAll lists.",  "data": [    {  "product\_no": "220309208600000040",  "product\_name": "product\_name0000",  "classift": "class\_0001",  "specification": "specification\_0000",  "unit": "支",  "remake": "Test0000",  "attribute1": "",  "attribute2": "",  "attribute3": "",  "attribute4": "",  "reserve": "555",  "createtime": "2023-02-07 22:48:01",  "updatetime": "2023-02-08 18:58:38"  },    {  "product\_no": "220309208600000041",  "product\_name": "product\_name0001",  "classift": "class\_0001",  "specification": "specification\_0001",  "unit": "支",  "remake": "Test0001",  "attribute1": "",  "attribute2": "",  "attribute3": "",  "attribute4": "",  "reserve": "",  "createtime": "2023-02-07 03:21:29",  "updatetime": "2023-02-08 18:58:38"  }  ]  } | |
|  | |

|  |  |
| --- | --- |
| 查詢品號ID | |
| 功能說明 | **以品號ID查詢資料** |
| url: <http://192.168.248.34:3000/catalog/product/220309208600000040>  Method: GET  req.params = 220309208600000040 | |
| Body:  無 | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "find product successfully",  "data": [    {  "product\_no": "220309208600000040",  "product\_name": "product\_name0000",  "classift": "class\_0001",  "specification": "specification\_0000",  "unit": "支",  "remake": "Test0000",  "attribute1": "",  "attribute2": "",  "attribute3": "",  "attribute4": "",  "reserve": "555",  "createtime": "2023-02-07 22:48:01",  "updatetime": "2023-02-08 18:58:38"  }  ]  }  res.status(200);  Response:查詢無資料  {  "service": 2,  "code": 404,  "msg": "Not found user\_id: user001",  "data": []  } | |
|  | |

|  |  |
| --- | --- |
| 更新品號資料 | |
| 功能說明 | **以品號ID更新資料** |
| url: <http://192.168.248.34:3000/catalog/product/220309208600000040>  Method: PUT  req.params = 220309208600000040 | |
| Body:  {  "product\_no": "220309208600000040",  "product\_name": "product\_name0000",  "classift": "class\_0001",  "specification": "specification\_0000",  "unit": "支",  "remake": "Test0000",  "attribute1": "",  "attribute2": "",  "attribute3": "",  "attribute4": "",  "reserve": ""  } | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "Product updateById product\_no: 220309208600000040 successfully.",  "data": []  }  res.status(200);  Response:無資料  {  "service": 2,  "code": 404,  "msg": "Not find product\_no: 220309208600000040.",  "data": []  } | |
|  | |

|  |  |
| --- | --- |
| 刪除品號資料 | |
| 功能說明 | **以品號ID刪除資料** |
| url: <http://192.168.248.34:3000/catalog/product/220309208600000040>  Method: DELETE | |
| Body:  無 | |
| res.status(200);  Response:有資料  {  "service": 1,  "code": 200,  "msg": "deleteOne successfully product\_no: 220309208600000040",  "data": []  }  res.status(200);  Response:無資料  {  "service": 2,  "code": 404,  "msg": "Delete not found product\_no: 220309208600000040",  "data": []  } | |
|  | |