

DataBase

- **Product Table**

ProductID	ProductName	isActive
1	UShop	1
2	RMM	1
3	ILPS	1

- **StageType Table**

TypeID	TypeName	Description	isActive
1	Prepare	准备	1
2	CodeAnalysis	对代码进行静态检查	1
3	Build	编译	1
4	Test	测试	1
5	Deploy	部署	1
6	Other	其他类别	1

- **Stage Table**

StageID	StageName	TypeID	ProductID	Platform	Source Code	EntryProgram	Parameters	isActive
1	C#代码检查	2	0	dockerimage	giturl	analysis.sh	{"paraName1":"","paraName2":""}	1
2	Java代码检查	2	0	dockerimage	giturl	analysis.sh	{"paraName1":"","paraName2":""}	1
3	Build	3	1	dockerimage	giturl	build.sh	{"paraName1":"","paraName2":""}	1
4	APITest	4	1	node(Linux)	giturl	apitest.py	{"paraName1":"value1","paraName2":"value4"}	1
5	UITest	4	1	node(windows)	giturl	uitest.exe	{"paraName1":"value1","paraName2":"value5"}	1

- **DevOpsPlan Table**

DevOpsPlanID	PlanName	ProductID	isActive
1	UShopApi	1	1
2	UShopMsgPro	1	1
3	RMMApi	2	1

- DevOps_Plan_Stage Table

DevOpsPlanID	StageID	PreStageIDs	isSync	Parameter
1	2	0	1	{"paraName1":"1","paraName2":"2"}
1	3	2	1	{"paraName1":"1","paraName2":"2"}
1	4	2,3	0	{"paraName1":"value1","paraName2":"value4"}
2	1	0	1	
2	2	1	0	
3	1	0	1	
3	2	0	1	
3	3	1	0	

- DevOpsBuild Table

BuildID	BuildIndex	DevOpsPlanID	BuildStatus	StartTime	EndTime
1	1	1	pass		
2	1	2	fail		
3	1	3			
4	2	2			

- DevOps_Build_Stage Table

BuildID	StageName	Status	Detail	StartTime	EndTime
1	Java 代码检查	notrun	{"url":"http://***"}	14000000	
1	Build	notrun			
1	APITest	notrun			
2	C#代码检查	pass			
2	Java 代码检查	fail			
3	C#代码检查	pass			
3	Java 代码检查	pass			
3	Build	running			
4	C#代码检查	notrun			
4	Java 代码检查	notrun			

红色为李洁正在实现的部分：

/TestCloud/DevOps/Product/list
/TestCloud/DevOps/Product/add
/TestCloud/DevOps/Product/update
/TestCloud/DevOps/StageType/list
/TestCloud/DevOps/StageType/add
/TestCloud/DevOps/StageType/update
/TestCloud/DevOps/Stage/list?ProductID=0&TypeID=1
/TestCloud/DevOps/Stage/add
/TestCloud/DevOps/Stage/update
/TestCloud/DevOps/Plan/list
/TestCloud/DevOps/Plan/add
/TestCloud/DevOps/Plan/update
/TestCloud/DevOps/Build/list
/TestCloud/DevOps/Build/exec
/TestCloud/DevOps/Build/update
/TestCloud/DevOps/Stage/info

Note: 加粗的参数为 required 的参数

/TestCloud/DevOps/Product/list	
GET	获取 Product 的列表
Response	<pre>{ "ErrorCode": 1000, "Description": "OK", "TotalCount": 2, "ProductList": [{ "ProductID": 1, "ProductName": "UShop" }, { "ProductID": 2, "ProductName": "RMM" }] }</pre>
/TestCloud/DevOps/Product/add	
POST	添加一个 product

Request	{ "ProductName": "UShop", }
Response	{ "ErrorCode": "1000", "Description": "OK", "ProductID": "1" }
/TestCloud/DevOps/Product/update	
POST	更新一个 product
Request	{ "ProductID": "1", "ProductName": "UShop" }
Response	{ "ErrorCode": "1000", "Description": "OK" }
/TestCloud/DevOps/StageType/list	
GET	获取 StageType 的列表
Response	{ "ErrorCode": "1000", "Description": "OK", "TotalCount": "2", "StageTypeList": [{ "TypeID": "1", "TypeName": "Prepare", "Description": "准备" }, { "TypeID": "2", "TypeName": "CodeAnalysis", "Description": "对代码进行静态检查" }] }
/TestCloud/DevOps/StageType/add	
POST	添加一个 StageType

Request	<pre>{ " TypeName":"单元测试", " Description ":"uint test" }</pre>
Response	<pre>{ "ErrorCode":"1000", "Description":"OK", "TypeID":"1" }</pre>
/TestCloud/DevOps/StageType/update	
POST	更新一个 StageType
Request	<pre>{ " TypeID ":"1", " TypeName":"prepare" , " Description ":"prepare" , " isActive":"0" }</pre> <p>除 id 外，至少有一项要更新的。</p>
Response	<pre>{ "ErrorCode":"1000", "Description":"OK" }</pre>
/TestCloud/DevOps/Stage/list?ProductID=0&TypeID=1	
GET	获取某个 Product 的某个 StageTypeID 下的 stage 的 list。 ProductID=0 的时候返回的是 common 支持的 stage， productID 非 0 是表示此 stage 为自定义的属于某个产品的。
Parameter	ProductID TypeID

Response	<pre> { "ErrorCode": "1000", "Description": "OK", "TotalCount": "2", "StageList": [{ "StageID": "1", "StageName": "C#代码检查", "Platform": "dokcerimage", "SourceCode": "http : ...git(giturl)", "EntryProgram": "analysis.sh", "Parameters": {"paraName1": "", "paraName2": ""}, }, { "StageID": "2", "StageName": "java 代码检查", "Platform": "dokcerimage", "SourceCode": "http : ...git(giturl)", "EntryProgram": "analysis.sh", "Parameters": {"paraName1": "", "paraName2": ""} }] } </pre>
/TestCloud/DevOps/Stage/add	
POST	添加一个 Stage
Request	<pre> { " StageName": "APITest", " TypeID": "1" , " ProductID": "1", " Plateform": "dockerimage", " SourceCode": "giturl", " EntryProgram": "1", " Parameters": {" paraName1": "aaa", " paraName2": "bbb"} } </pre>
Response	<pre> { "ErrorCode": "1000", "Description": "OK", "StageID": "1" } </pre>
/TestCloud/DevOps/Stage/update	

POST	更新一个 Stage
Request	<pre>{ "StageID": "1" " StageName": "APITest", " TypeID": "1" , " ProductID": "1", " Plateform": "dockerimage", " SourceCode": "giturl", " EntryProgram": "1", " Parameters": { " paraName1": "aaa", " paraName2": "bbb"} " isActive": "0" }</pre> <p>除 ID 外，至少有一项要更新的</p>
Response	<pre>{ "ErrorCode": "1000", "Description": "OK" }</pre>
/TestCloud/DevOps/Plan/list	
GET	获取某个 Product 下对应的 DevOpsPlan 的 list
Parameter	ProductID
Response	<pre>{ "ErrorCode": 1000, "Description": "OK", "TotalCount": 2, "PlanList": [{ "DevOpsPlanID": 1, "PlanName": "UShopApi", }, { "DevOpsPlanID": 2, "PlanName": "UShopMsgPro", }] }</pre>
/TestCloud/DevOps/Plan/add	
POST	添加一个 DevOps plan

Request	<pre> { "PlanName": "UShopAPI", "ProductID": "1", "StageCount": "2", "StageList": [{ "StageId": "1", "PreStageIDs": "0", "isSync": "1", "Parameters": { "paraName1": "aaa", "paraName2": "bbb" } }, { "StageId": "2", "PreStageIDs": "1", "isSync": "1", "Parameters": { "paraName1": "aaa", "paraName2": "bbb" } }] } </pre>
Response	<pre> { "ErrorCode": "1000", "Description": "OK", "DevOpsPlanID": "1" } </pre>
/TestCloud/DevOps/Plan/update	
POST	更新一个 DevOps plan
Request	<pre> { "DevOpsPlanID": "1", "PlanName": "UShopAPI", "ProductID": "1", "isActive": "1", "StageCount": "2", "StageList": [{ "StageId": "1", "PreStageIDs": "0", "isSync": "1", "Parameters": { "paraName1": "aaa", "paraName2": "bbb" } }, { "StageId": "2", </pre>

	<pre> "PreStageIDs":"1", "isSync":"1", "Parameters":{" paraName1":"aaa", " paraName2":"bbb"} }] } </pre> <p>除 id 外至少要有一项要更新的。</p>
Response	<pre> { "ErrorCode":"1000", "Description":"OK", } </pre>
/TestCloud/DevOps/Build/list	
GET	获取某个 DevOpsPlan 下对应的 Build list
Parameter	DevOpsPlanID
Response	<pre> { "ErrorCode": 1000, "Description":"OK", "TotalCount": 2, "BuildList":[{ "BuildID": 1, "BuildIndex": 1, "Status":"Pass", "StartTime": 1491904800, "EndTime": 1491904800 }, { "BuildID": 2, "BuildIndex": 2, "Status":"Fail", "StartTime ": 1491904800, "EndTime": 1491904800 }] } </pre>
/TestCloud/DevOps/Build/exec	

POST	添加一个 Build 并触发 Jenkins 进行执行
Request	<pre>{ "DevOpsPlanID": 1 "JobName":"Test", (对应数据库中 PlanName) "Parameter": { "parameter": [{ "name":"BuildID", "value":"1" }] } }</pre>
Response	<pre>{ "ErrorCode": 1000, "Description":"OK" }</pre>
/TestCloud/DevOps/Stage/list	
GET	获取某个 Build 对应的 Stage
Parameter	BuildID
Response	<pre>{ "ErrorCode": 1000, "Description":"OK", "StageCount": 2, "StageList":[{ "StageID": 1, "StageName":""," "PreStageID":["1"], "Status":"Pass", "Detail":{}, "StartTime": 1400000, "EndTime": 1400000 }, { "StageID": 2,</pre>

	<pre> "StageName ":"", "PreStageID":[1], >Status:"Fail", "Detail":{}, "StartTime": 1400000, "EndTime": 1400000 }] } </pre>
/TestCloud/DevOps/Stage/update	
POST	修改某个 Stage 的状态
Request	<pre> { "BuildID": 1, "StageName":"Build", "Status":"running" , "Detail":{"URL":"http://***"} } </pre>
Response	<pre> { "ErrorCode": 1000, "Description":"OK" } </pre>
/TestCloud/DevOps/Build/update	
POST	修改某个 Build 的状态
Request	<pre> { "BuildID": 1, "BuildStatus":"running" } </pre>
Response	<pre> { "ErrorCode": 1000, "Description":"OK" } </pre>