

HSU Enterprise Application Stack(HSU EAS)

MANUAL

Marvin
HSU FORUM
版本 1.0.5

目录

1. 前言.....	2
2. 使用步骤.....	2
3. CRUD 循序机制.....	4
3.1 查询	4
3.2 新增	5
3.3 修改 & 删除.....	6
4. 功能.....	7
4.1 功能项目	7
4.2 系统功能	9
4.3 系统群组	10
4.4 系统账号	12
4.5 系统模块	13
5. 其他.....	15
5.1 default-faces-config.xml	15

1. 前言

HSU Enterprise Application Stack, 这是 Marvin 的拙作, 喜欢的人可以到 Git Hub 下载来使用。主要是堆栈主流的 Java 技术, 包含 Prime Faces(JSF)、Spring、Hibernate(JPA)与 Spring Security, 用途主要来加速企业内部的应用程序开发, 与使用 Java script 的技术开发相比, 约可增加 20%~30% 的开发速度。HSU Common 为共享组件, HSU EAS 为应用程序主体。默认的账号与密码为账号: admin、密码: abcd1234

- 支持语言为繁体中文、简体中文与英文
- 支援 DB 有 Oracle、SQL Server 与 MySQL
- 支援 AP Server 有 Tomcat、Jetty、JBoss WildFly、JBoss EAP 与 WebLogic。
- HSU Common Git Hub 下载位置为 <https://github.com/MarvinHsu/HSUCommon>
- HSU EAS Git Hub 下载位置为 <https://github.com/MarvinHsu/EAS>
- Demo war 下载位置为 (Google Drive)

<https://drive.google.com/drive/folders/0B2s05eiH2k6CYXJ0LTmNVBUVXM>

P.S. Tomcat 请设定 jndi datasource 名称为 jndi/EASDS. JBoss 请设定 jndi datasource 名称为 java:/EASXADS. Weblogic 请设定 jndi datasource 名称为 jndi/EASXADS

- 支援的 Java AP Server

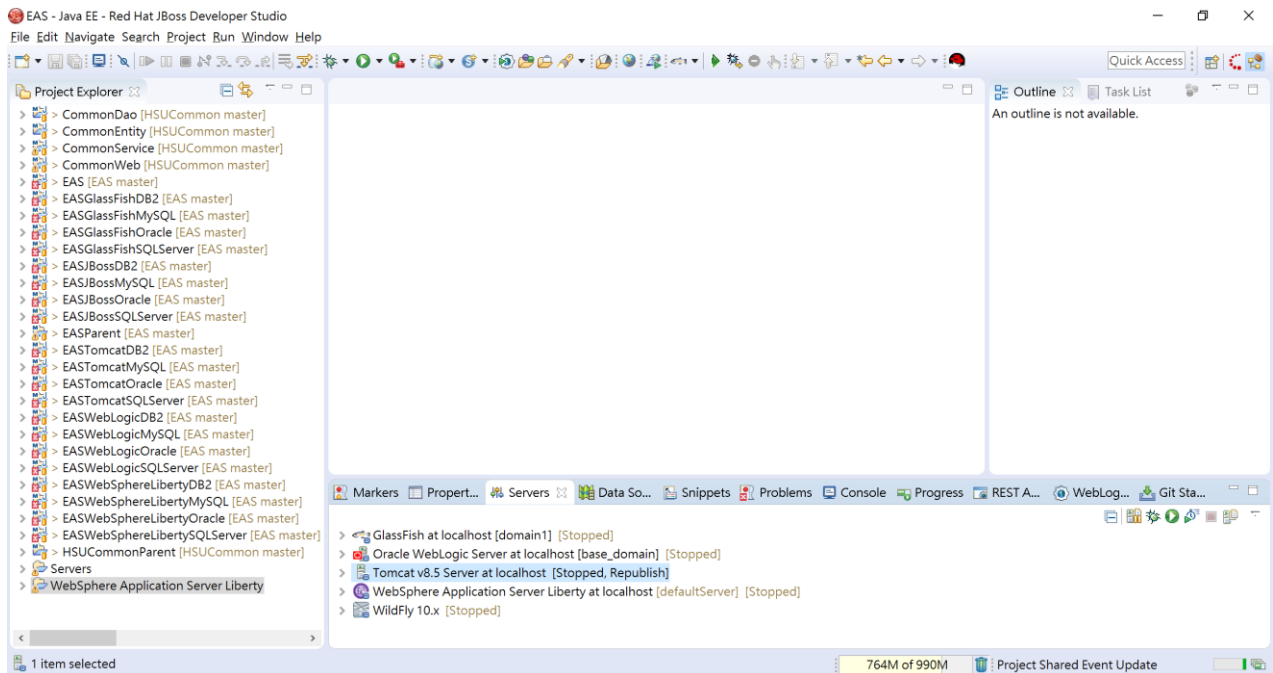
Java AP Server	版本
Tomcat	7+
Jetty	9.4.12+
JBoss EAP	7+
JBoss WildFly	10+
WebLogic	12.2.1.3+
WebSphere Liberty	17.0.0.2+
Payara(GlassFish)	4.1.1.164+

- 支援的 Data Base

Data Base	版本
MySQL	5.5+
Oracle	10g+
SQL Server	2008+
DB2	9.7+
PostgreSQL	9.6+

2. 使用步骤

1. 在下载后可至 Eclipse 汇入 Git Maven Project 后, 如以下截图



2. 依数据库类型与环境执行 DB Schema SQL 与 DB Init Data SQL

- ▼ DB Init Data SQL
 - Init Data en.sql
 - Init Data zh_CN.sql
 - Init Data zh_TW.sql
- ▼ DB Schema SQL
 - DB2 Schema.sql
 - MySQL Schema.sql
 - Oracle Schema.sql
 - PostgreSQL Schema.sql
 - SQL Server Schema.sql

3. 启动 Server, 并浏览网页, 即可进入登入页面。默认的账号与密码为账号: admin、密码: abcd1234

HSU EAS

red
Change skin
Logout

Please enter account and password

account

password

remember me ☐

Login
reset

4. 登入后即可进入管理页面操作基本功能

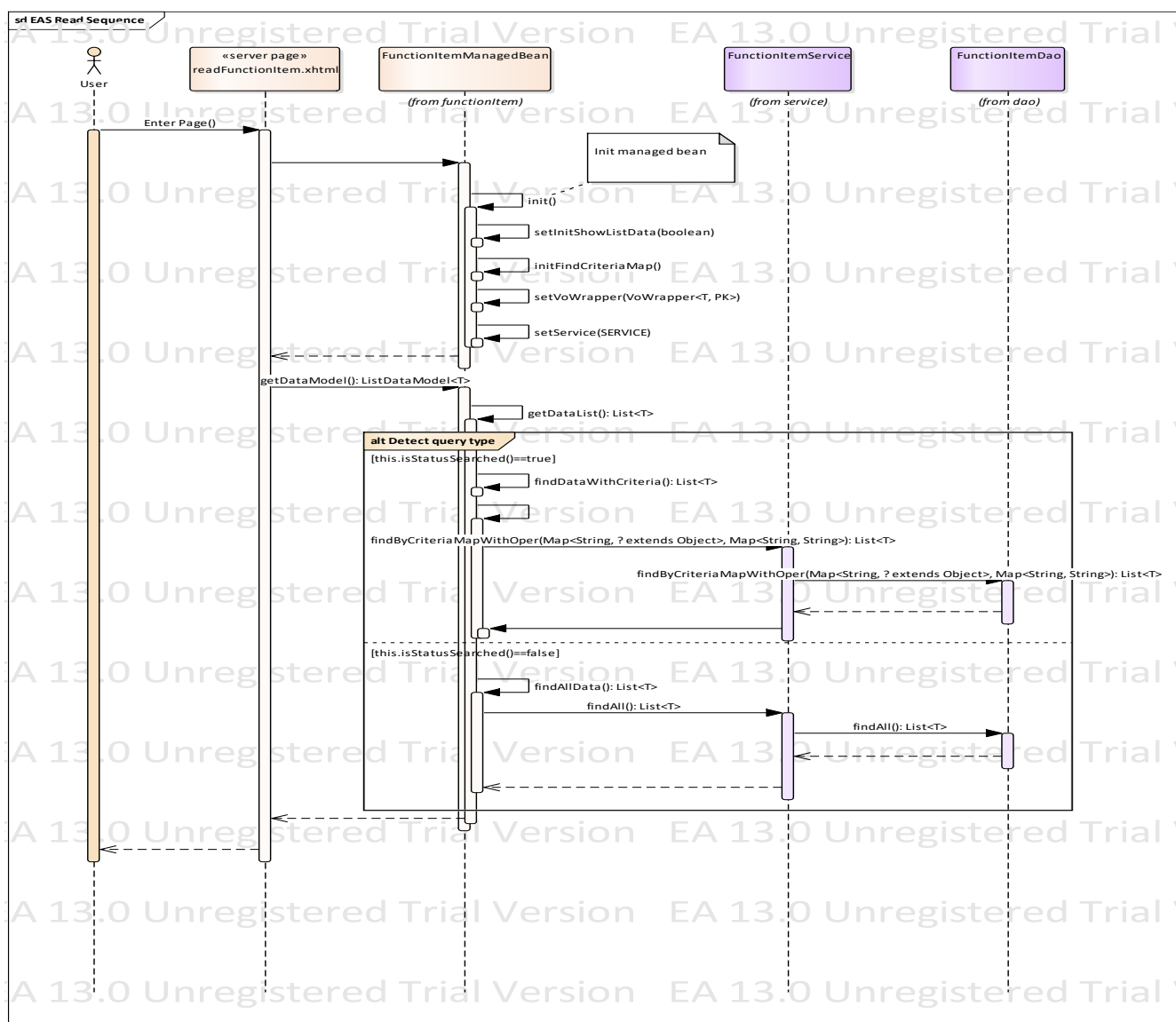
Management
Module

Welcome To HSU Enterprise Application Stack

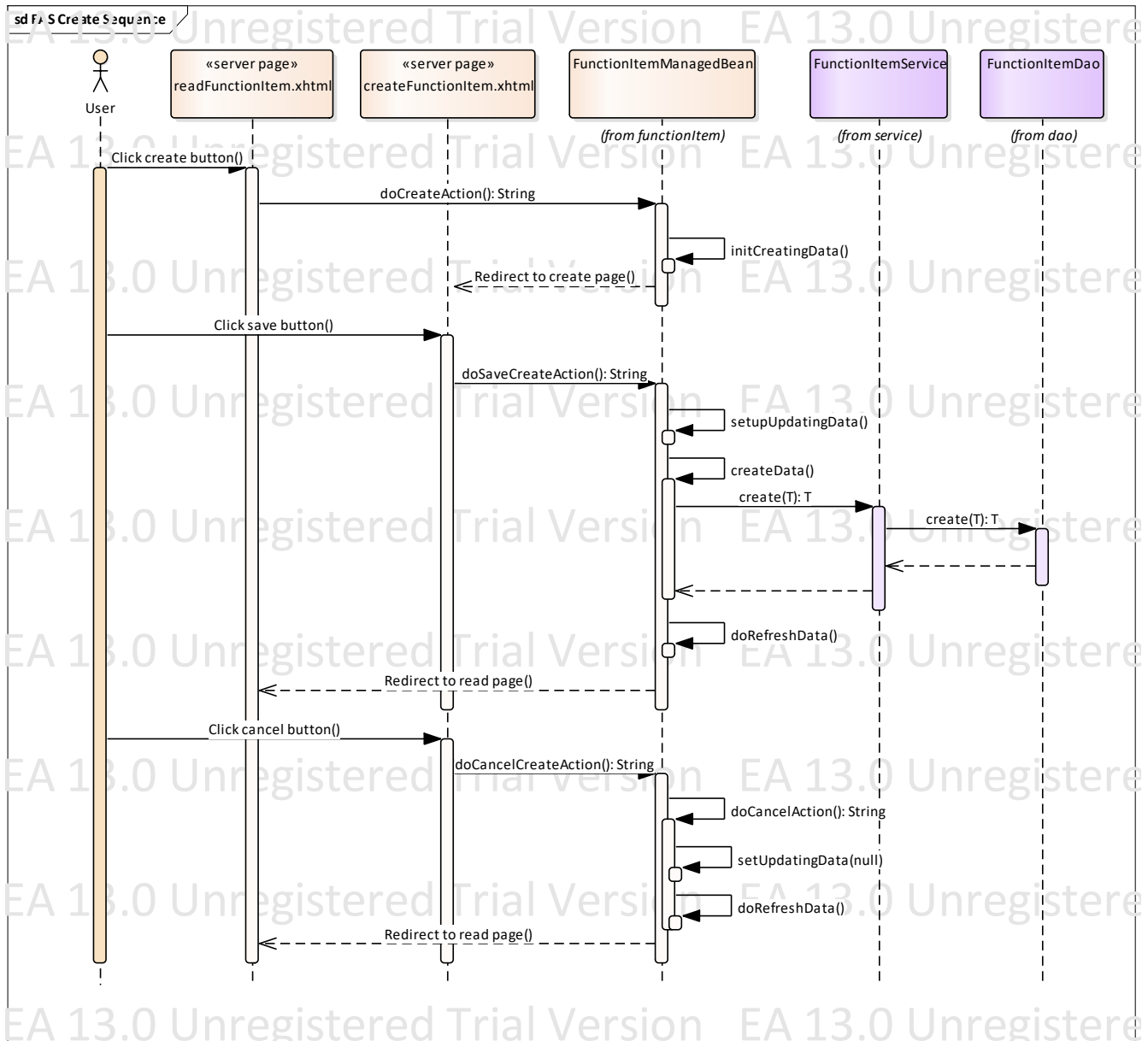
Function item Config
Function Config
Group Config
User Config
Module Config

3. CRUD 循序机制

3.1 查询



3.2 新增



```

sequenceDiagram
    actor User
    participant ServerPage as «server page»  
readFunctionItem.xhtml
    participant ServerPage2 as «server page»  
updateFunctionItem.xhtml
    participant FIMB as FunctionItemManagedBean
    participant FIS as FunctionItemService
    participant FID as FunctionItemDao

    Note over FIS: init update data
    FIS-->>FIM: (from service)  
FIM->>FIM: setUpUpdatingData(ValueObject<T, PK>)  
FIM->>FIM: initUpdatingData(ValueObject<FunctionItem, java.lang.String>)  
FIM->>ServerPage2: Redirect to update page()

    User->>ServerPage: Click row to update()
    ServerPage->>FIM: doUpdateAction(): String
    FIM->>FIS: update(T): T
    FIS-->>FIM: 
    FIM->>ServerPage2: doRefreshData()
    ServerPage2-->>User: Redirect to read page()

    User->>ServerPage2: Click save button()
    ServerPage2->>FIM: doSaveUpdateAction(): String
    FIM->>FIM: setupUpdatingData()
    FIM->>FIM: updateData()
    FIM->>FIS: update(T): T
    FIS-->>FIM: 
    FIM->>ServerPage2: doRefreshData()
    ServerPage2-->>User: Redirect to read page()

    User->>ServerPage2: Click cancel button()
    ServerPage2->>FIM: doCancelUpdateAction(): String
    FIM->>FIM: doCancelAction(): String
    FIM->>FIM: doRefreshData()
    FIM-->>ServerPage2: Redirect to read page()

    User->>ServerPage2: Click delete button()
    ServerPage2->>FIM: doDeleteAction(): String
    FIM->>FIM: deleteOneData(T)
    FIM->>FIS: delete(T)
    FIS->>FID: delete(T)
    FIS-->>FIM: 
    FIM->>FIM: doRefreshData()
    FIM-->>ServerPage2: Redirect to read page()
  
```

The diagram illustrates the sequence of interactions for updating and deleting function items. It involves a User, two server pages, and three system components: FunctionItemManagedBean, FunctionItemService, and FunctionItemDao.

Update Sequence:

- The User clicks a row to update, triggering `doUpdateAction(): String` from the «server page» `readFunctionItem.xhtml` to `FunctionItemManagedBean`.
- `FunctionItemManagedBean` receives `init update data` from `FunctionItemService` (via `(from service)`) and performs `setUpUpdatingData(ValueObject<T, PK>)` and `initUpdatingData(ValueObject<FunctionItem, java.lang.String>)` internally.
- `FunctionItemManagedBean` then sends `update(T): T` to `FunctionItemService`, which returns a response.
- `FunctionItemManagedBean` performs `doRefreshData()` and sends `Redirect to read page()` back to the «server page».
- The «server page» then sends `Redirect to read page()` to the User.

Save Sequence:

- The User clicks the save button, triggering `doSaveUpdateAction(): String` from the «server page» `updateFunctionItem.xhtml` to `FunctionItemManagedBean`.
- `FunctionItemManagedBean` performs `setupUpdatingData()` and `updateData()` internally.
- `FunctionItemManagedBean` sends `update(T): T` to `FunctionItemService`, which returns a response.
- `FunctionItemManagedBean` performs `doRefreshData()` and sends `Redirect to read page()` back to the «server page».
- The «server page» then sends `Redirect to read page()` to the User.

Cancel Sequence:

- The User clicks the cancel button, triggering `doCancelUpdateAction(): String` from the «server page» `updateFunctionItem.xhtml` to `FunctionItemManagedBean`.
- `FunctionItemManagedBean` performs `doCancelAction(): String` and `doRefreshData()` internally.
- `FunctionItemManagedBean` sends `Redirect to read page()` back to the «server page».
- The «server page» then sends `Redirect to read page()` to the User.

Delete Sequence:

- The User clicks the delete button, triggering `doDeleteAction(): String` from the «server page» `updateFunctionItem.xhtml` to `FunctionItemManagedBean`.
- `FunctionItemManagedBean` performs `deleteOneData(T)` internally.
- `FunctionItemManagedBean` sends `delete(T)` to `FunctionItemService`, which then sends `delete(T)` to `FunctionItemDao`.
- `FunctionItemService` returns a response to `FunctionItemManagedBean`.
- `FunctionItemManagedBean` performs `doRefreshData()` and sends `Redirect to read page()` back to the «server page».
- The «server page» then sends `Redirect to read page()` to the User.

4. 功能

4.1 功能项目

Function item

Name

Code

URL

Find(Alt+q)

Create(Alt+c)

(1 of 1)

1

25

ID	Name	Code	URL
10000001-0000-0000-0000-000000000000	BackOffice	default	/default.jsf
10000001-0000-0000-0000-000000000001	Query function item	query	/kernel/item/readItem.jsf
10000001-0000-0000-0000-000000000002	Create function item	create	/kernel/item/createItem.jsf
10000001-0000-0000-0000-000000000003	Update function item	update	/kernel/item/updateItem.jsf
10000001-0000-0000-0000-000000000004	Delete function item	delete	/kernel/item/updateItem.jsf
10000001-0000-0000-0000-000000000005	Query function	query	/kernel/function/readFunction.jsf
10000001-0000-0000-0000-000000000006	Create function	create	/kernel/function/createFunction.jsf
10000001-0000-0000-0000-000000000007	Update function	update	/kernel/function/updateFunction.jsf
10000001-0000-0000-0000-000000000008	Delete function	delete	/kernel/function/updateFunction.jsf
10000001-0000-0000-0000-000000000009	Query group	query	/kernel/group/readGroup.jsf
10000001-0000-0000-0000-000000000010	Create group	create	/kernel/group/createGroup.jsf
10000001-0000-0000-0000-000000000011	Update group	update	/kernel/group/updateGroup.jsf
10000001-0000-0000-0000-000000000012	Delete group	delete	/kernel/group/updateGroup.jsf
10000001-0000-0000-0000-000000000013	Query user	query	/kernel/user/readUser.jsf
10000001-0000-0000-0000-000000000014	Create user	create	/kernel/user/createUser.jsf
10000001-0000-0000-0000-000000000015	Update user	update	/kernel/user/updateUser.jsf
10000001-0000-0000-0000-000000000016	Delete user	delete	/kernel/user/updateUser.jsf
10000001-0000-0000-0000-000000000017	Query module	query	/kernel/module/readModule.jsf
10000001-0000-0000-0000-000000000018	Create module	create	/kernel/module/createModule.jsf
10000001-0000-0000-0000-000000000019	Update module	update	/kernel/module/updateModule.jsf
10000001-0000-0000-0000-000000000020	Delete module	delete	/kernel/module/updateModule.jsf

(1 of 1)

1

25

- 输入查询条件，按下查询可查询数据
- 按下建立可建立一笔功能项目数据
- 点选名称可进入修改画面

Function item-Create

ID	10569e6f-8b05-45fa-9a9a-6
Name	
Code	
URL	
Function	-----▼

Save(Alt+s)Cancel(Alt+r)

- 输入数据，按下储存，可建立一笔功能项目数据
- 按下取消，回到查询页面

Function item-Update

ID	10000001-0000-0000-0000
Name	Query function item
Code	query
URL	/kernel/item/readItem.jsf
Function	Function item Config▼

Delete(Alt+d)Save(Alt+s)Cancel(Alt+r)

- 按下删除，可删除整笔功能项目
- 输入数据，按下储存，可修改整笔功能项目数据
- 按下取消，回到查询画面

4.2 系统功能

Function

Name

code

Outcome

Find(Alt+q)

Create(Alt+c)

(1 of 1)

1

25

ID	Name	code	Outcome	Create date	Update date
10000002-0000-0000-0000-000000000001	ROOT	ROOT		2018/11/25	
10000002-0000-0000-0000-000000000002	Function item Config	functionItem	functionItem	2018/11/25	
10000002-0000-0000-0000-000000000003	Function Config	function	function	2018/11/25	
10000002-0000-0000-0000-000000000004	Group Config	group	group	2018/11/25	
10000002-0000-0000-0000-000000000005	User Config	user	user	2018/11/25	
10000002-0000-0000-0000-000000000006	Module Config	module	module	2018/11/25	

(1 of 1)

1

25

- 输入查询条件，按下查询可查询数据
- 按下建立可建立一笔功能数据
- 点选名称可进入修改画面

Function-Create

ID

87ab04b1-0fc4-469a-a347~

Name

code

Outcome

Module

Save(Alt+s)

Cancel(Alt+r)

- 输入数据，按下储存，可建立一笔功能数据
- 按下取消，回到查询页面

Function-Update

ID	10000002-0000-0000-0000
Name	Function item Config
code	functionItem
Outcome	functionItem
Module	Management Module ▼

Delete(Alt+d) Save(Alt+s) Cancel(Alt+r)

- 按下删除，可删除整笔功能
- 输入数据，按下储存，可修改整笔功能数据
- 按下取消，回到查询画面

4.3 系统群组

Group

Name	
Code	
Find(Alt+q) Create(Alt+c)	

(1 of 1) 1 25 ▼		
ID ↕	Name ↕	Code ↕
10000005-0000-0000-0000-000000000001	Management	MANAGE
(1 of 1) 1 25 ▼		

- 输入查询条件，按下查询可查询数据
- 按下建立可建立一笔群组数据
- 点选名称可进入修改画面

Group-Create

ID

5fe37d7e-1d78-4274-83cd-

Name

Code

Group function

ID	Name			
ROOT	<input type="checkbox"/> BackOffice			
Function item Config	<input type="checkbox"/> Delete function item	<input type="checkbox"/> Update function item	<input type="checkbox"/> Create function item	<input type="checkbox"/> Query function item
Function Config	<input type="checkbox"/> Delete function	<input type="checkbox"/> Update function	<input type="checkbox"/> Create function	<input type="checkbox"/> Query function
Group Config	<input type="checkbox"/> Query group	<input type="checkbox"/> Update group	<input type="checkbox"/> Create group	<input type="checkbox"/> Delete group
User Config	<input type="checkbox"/> Update user	<input type="checkbox"/> Create user	<input type="checkbox"/> Query user	<input type="checkbox"/> Delete user
Module Config	<input type="checkbox"/> Delete module	<input type="checkbox"/> Update module	<input type="checkbox"/> Create module	<input type="checkbox"/> Query module

Save(Alt+s)

Cancel(Alt+r)

- 输入数据，按下储存，可建立一笔群组数据
- 按下取消，回到查询页面

Group-Update

ID

10000005-0000-0000-0000

Name

Management

Code

MANAGE

Group function

ID	Name			
ROOT	<input checked="" type="checkbox"/> BackOffice			
Function item Config	<input checked="" type="checkbox"/> Delete function item	<input checked="" type="checkbox"/> Update function item	<input checked="" type="checkbox"/> Create function item	<input checked="" type="checkbox"/> Query function item
Function Config	<input checked="" type="checkbox"/> Delete function	<input checked="" type="checkbox"/> Update function	<input checked="" type="checkbox"/> Create function	<input checked="" type="checkbox"/> Query function
Group Config	<input checked="" type="checkbox"/> Query group	<input checked="" type="checkbox"/> Update group	<input checked="" type="checkbox"/> Create group	<input checked="" type="checkbox"/> Delete group
User Config	<input checked="" type="checkbox"/> Update user	<input checked="" type="checkbox"/> Create user	<input checked="" type="checkbox"/> Query user	<input checked="" type="checkbox"/> Delete user
Module Config	<input checked="" type="checkbox"/> Delete module	<input checked="" type="checkbox"/> Update module	<input checked="" type="checkbox"/> Create module	<input checked="" type="checkbox"/> Query module

Delete(Alt+d)

Save(Alt+s)

Cancel(Alt+r)

- 按下删除，可删除整笔功能
- 输入数据，按下储存，可修改整笔群组数据
- 按下取消，回到查询画面

4.4 系统账号

User

Name

E-Mail

Account

Find(Alt+q)

Create(Alt+c)

(1 of 1)

1

25

ID	Name	E-Mail	Account	Enable	Create date	Update date
10000006-0000-0000-0000-000000000001	ADMIN	abc@sample.com	ADMIN	true	2018/11/25	

(1 of 1)

1

25

- 输入查询条件，按下查询可查询数据
- 按下建立可建立一笔账号数据
- 点选名称可进入修改画面

User-Create

ID

165dcac8-0c2c-4387-bf94-4

Name

Password

E-Mail

Account

Enable

Group

	ID	Name
<input type="checkbox"/>	10000005-0000-0000-0000-000000000001	Management

Save(Alt+s)

Cancel(Alt+r)

- 输入数据，按下储存，可建立一笔账号数据
- 按下取消，回到查询页面

User-Update

ID	10000006-0000-0000-0000
Name	ADMIN
Password	*****
E-Mail	abc@sample.com
Account	ADMIN
Enable	<input checked="" type="checkbox"/>

Group		
<input checked="" type="checkbox"/>	ID	Name
<input checked="" type="checkbox"/>	10000005-0000-0000-0000-000000000001	Management

Delete(Alt+d)Save(Alt+s)Cancel(Alt+r)

- 按下删除，可删除整笔功能
- 输入数据，按下储存，可修改整笔账号数据
- 按下取消，回到查询画面

4.5 系统模块

Module

Name	
Code	
<div>Find(Alt+q)Create(Alt+c)</div>	

(1 of 1)		
id ↕	Name ↕	Code ↕
10000003-0000-0000-0000-000000000001	Management Module	SYSTEM
(1 of 1)		

- 输入查询条件，按下查询可查询数据
- 按下建立可建立一笔模块数据
- 点选名称可进入修改画面

Module-Create

id	11aeda1f-8d54-4a4b-bfcb-8
Name	
Code	

Function		
<input type="checkbox"/>	ID	Name
<input type="checkbox"/>	10000002-0000-0000-0000-000000000001	ROOT
<input type="checkbox"/>	10000002-0000-0000-0000-000000000002	Function item Config
<input type="checkbox"/>	10000002-0000-0000-0000-000000000003	Function Config
<input type="checkbox"/>	10000002-0000-0000-0000-000000000004	Group Config
<input type="checkbox"/>	10000002-0000-0000-0000-000000000005	User Config
<input type="checkbox"/>	10000002-0000-0000-0000-000000000006	Module Config

Save(Alt+s)

Cancel(Alt+r)

- 输入数据，按下储存，可建立一笔模块数据
- 按下取消，回到查询页面

Module-Update

id	10000003-0000-0000-0000
Name	Management Module
Code	SYSTEM

Function		
<input type="checkbox"/>	ID	Name
<input type="checkbox"/>	10000002-0000-0000-0000-000000000001	ROOT
<input checked="" type="checkbox"/>	10000002-0000-0000-0000-000000000002	Function item Config
<input checked="" type="checkbox"/>	10000002-0000-0000-0000-000000000003	Function Config
<input checked="" type="checkbox"/>	10000002-0000-0000-0000-000000000004	Group Config
<input checked="" type="checkbox"/>	10000002-0000-0000-0000-000000000005	User Config
<input checked="" type="checkbox"/>	10000002-0000-0000-0000-000000000006	Module Config

Delete(Alt+d)

Save(Alt+s)

Cancel(Alt+r)

- 按下删除，可删除整笔功能
- 输入数据，按下储存，可修改整笔模块数据
- 按下取消，回到查询画面

5. 其他

5.1 default-faces-config.xml

default-faces-config.xml 有几个系统的全局设定

- systemCode 默认为 EAS，为系统识别代码
- devMode 预设为 true，可控制查询页面 ID 字段是否显示
- rowsOfPerPage 预设为 25，可控制查询页面一页的显示笔数
- timeZone 默认为 Asia/Taipei，页面的时区显示