**FOXCONN**

**CABG/SDS/SW**

**Subject: Release Control System Design Specification**

**Version: 0.9**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | | **Authors:** | Henry DY.Liu  [Henry.dy.liu@mail.foxconn.com](mailto:Henry.dy.liu@mail.foxconn.com) | | |
| **Revision History** | | | | | | | | |
| **Version** | **Date** | | **Status** | **Author** | | | **Reason for Reissue** |
| 0.9 | January 10, 2017 | | Initial Draft | Henry DY.Liu | | | Initial document. |
|  |  | |  |  | | |  |
|  |  | |  |  | | |  |
|  |  | |  |  | | |  |

**TABLE OF CONTENTS**

(This part should be automatically generated)

Contents

[1. Introduction 4](#_Toc425779471)

[1.1 Purpose 4](#_Toc425779472)

[1.2 Terminology 4](#_Toc425779473)

[2 Scope 4](#_Toc425779474)

[2.1 Requirements Covered 4](#_Toc425779475)

[2.2 Requirements Not Covered 5](#_Toc425779476)

[2.3 Hardware / Software Components For Develop the Product 5](#_Toc425779477)

[3 High-level Design description 5](#_Toc425779478)

[3.1 Design Overview 5](#_Toc425779479)

[4 Interfaces and Implemtation 6](#_Toc425779480)

[4.1 User Interface 6](#_Toc425779485)

[4.1.1 Login Page 6](#_Toc425779486)

[4.1.2 Change Password Dialog 7](#_Toc425779487)

[4.1.3 Home Page（Normal User） 9](#_Toc425779488)

[4.1.4 Transfer Asset Dialog 13](#_Toc425779489)

[4.1.5 Delete Asset Dialog 17](#_Toc425779490)

[4.1.6 Sign Process Dialog 18](#_Toc425779491)

[4.1.7 Home Page ( supervisor ) 21](#_Toc425779492)

[4.1.8 Sign Page 22](#_Toc425779493)

[4.1.9 Sign Confirm Dialog 24](#_Toc425779494)

[4.1.10 Home Page(Administrator) 25](#_Toc425779495)

[4.1.11 Add Asset Dialog 26](#_Toc425779496)

[4.1.12 Modify Asset Dialog 28](#_Toc425779497)

[4.1.13 System Configuration Page 30](#_Toc425779498)

[4.1.14 User Management Page 30](#_Toc425779499)

[4.1.15 Add User Dialog 32](#_Toc425779500)

[4.1.16 Modify User Dialog 34](#_Toc425779501)

[4.1.17 Delete User Dialog 37](#_Toc425779502)

[4.1.18 Home Page (HR User) 39](#_Toc425779503)

[4.2 Internal Interface 39](#_Toc425779504)

[4.2.1 MultiLanguage 39](#_Toc425779505)

[5 Other Design Considerations and IMPACT 40](#_Toc425779506)

[6 OPEN ISSUE and EFFORT 40](#_Toc425779507)

1. Introduction

This document covers the design and features of project added or modified in the product Release Control System 1.0 release.

* 1. Purpose
  2. Terminology

Terminology and acronyms are listed below (in alphabetical order)

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Normal User | Release Applier/Viewer/Reviewer/Approver |
| Administrator | System administrator |
| Release Applier/Viewer | User have privilege of apply a release, release query and file download. |
| Release Reviewer | User who can review a release. |
| Release Approver | User who can approve a release. |
|  |  |
|  |  |

* 1. References

List any associated or external reference document if any.

|  |  |  |  |
| --- | --- | --- | --- |
| **Title** | **Version** | **Author(s)** | **Comment** |
|  |  |  |  |
|  |  |  |  |

1. Scope

This document covers the design of Asset Management System 1.0.

* 1. Requirements Covered

List requirement items under based on SW Requirement Spec.

Requirement ID1, ID2, etc.

|  |  |  |
| --- | --- | --- |
| **Requirement Item** | **Description** | **Detail** |
| Requirement ID1 | Developers can login to system to record release information, including Release version, Release date, Release owner, customer info(for which customer), release note(fix what issues), code reviewer, Source code SVN location, released package(BIN file and other files),etc | 1. Everyone can apply to release all projects. 2. Release team will be set to developer’s team who apply the release. 3. Developer can cancel a release before being reviewed. |
| Requirement ID2 | Each release must be reviewed and approved by release reviewer and Eric, if approved, then the release data will be stored into database and the released FW/SW will be uploaded to Server storage | 1. Release will be uploaded to ftp server after being approved. 2. Use current “team/project/release” path to store releases. 3. Release files should be packed to tar/tgz/zip/rar/7z before upload. |
| Requirement ID3 | Provide search/list feature, user can find one version release information quickly by input keywords, such as, by Employee ID, by customer, and we can view the release note to quickly check what changes in that version | 1. Everyone can query/download release files. 2. ？？ |

## Requirements Not Covered

* 1. Hardware / Software Components For Develop the Product

List Hardware/Software to the design is dependent on.

The following table has the software components that are used to develop the product:

| **Software** | **Provider** | **Version** |
| --- | --- | --- |
| Visual Studio Code | [code.visualstudio.com](http://www.baidu.com/link?url=YskmhLZMrNMePHONOWb4kulofiimPb7ogmmDlobTEviVLOCQb0KuMS41i5Ksbyx7&wd=&eqid=d6291729000222b200000004586dfc77) |  |
| Nodejs | nodejs.org |  |
| MySQL | www.mysql.com |  |

1. High-level Design description

## Design Overview

System Software architecture block diagram

FTP Server

Html5 / Ajax

NodeJS

MySQL Database

Function block diagram

Release Control System

Release File Management

Release Process/History Management

System Configuration

File Upload

Release Apply/Cancel

User Management

File Download

Release Review

Role Management

Release Approve

Team Management

Release Query

Project Management

1. Interfaces and Implemtation

Provide detail info about design/code interface, flow diagram.

All user interfaces are function indicative only.



## User Interface

### Login Page

软体发布系统

用户名:

密码:

登入

In this page, user can input username and password to login.

Note: Default password is employee number, User need to change password at the first login.

Internal Interfaces:

1. Purpose: User Login

Client: Login

Input:

Username

Password

Output:

Result Whether there is an error

Valid Whether user is valid

LoginInfo Information of a login

Server**:** POST /api/login

Route: app.post(“/api/login”)

Controller: UserController/login

Model: Login/post

DAO: dbhelper/query

DB: t\_user

### Change Password Dialog

修改密码

新密码:

确认密码:

取消

确定

User need to change password at the first login.

Internal Interfaces:

1. Purpose: Change Password.

Client: ChangePwd:

Input:

Password

Output:

Result Whether there is an error

Server**:** POST /api/user/:id

Route: app.post(“/api/user”)

Controller: UserController/ChangePwd

Model: User/save

DAO: dbhelper.query

DB: t\_user

### Home Page（Normal User）

|  |  |  |
| --- | --- | --- |
| 软体发布系统 | | 工号 姓名 登出 |
| 发布清单 |  | |
| 我的申请 |  | |
| 我的签核（签核数） |  | |
| Copyright 2015 Foxconn CABG ©. All Rights Reserved. | | |

In Home Page, user can：

1. Check Log information.
2. Logout.
3. Select different sub menu.

Internal Interfaces:

1. Purpose: User Logout.

Client: Logout:

Input:

ID Identify a user

Output:

Result Whether there is an error

Server: DELETE /api/login/:id

Route: app.delete(“api/login”)

Controller: UserController/Logout

Model: login/delete

1. Purpose: Get User Sign Count

Client: GetSignCnt:

Input:

ID Identify a user

Output:

Result Whether there is an error

Count User Sign Count

Server: GET /api/release

Route: app.get(“api/release”)

Controller: ReleaseController/GetSignCnt

Model: release/get

### Release List Sub Page

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 发布清单 | | | | | | | | | | | | |
| 课 工号 项目 客户 申请时间 核准时间 状态 | | | | | | | | | | 查询 导出 | | |
| 课 | 工号 | 姓名 | 项目 | 版本 | 客户 | 注释 | Svn | 申请时间 | 核准时间 | | 状态 | 位置 |
| bmc | \* | \* | test | 1.0 | test | \* | \* | \* | \* | | 核准 | \* |
|  |  |  |  |  |  |  |  |  |  | |  |  |
|  |  |  |  |  |  |  |  |  |  | |  |  |
| 首页 下一页 上一页 尾页 | | | | | | | | | | | | |

In this page, user can：

1. View all releases.
2. Export release list.
3. Query releases by team/work no/project/customer.
4. Download a release package which has been approved.

Note : The page initially shows all releases.

Internal Interfaces:

1. Purpose: Get Release List

Client: GetReleaseLst:

Input:

Params Team, work\_no, project, version, apply date, approve date, status

Output:

Result Whether there is an error

ReleaseList All Releases in accordance with params

Server**:** GET /api/release

Route: app.get(“/api/release”)

Controller: ReleaseController/GetReleaseLst

Model: Release/get

DAO: dbhelper.query

DB: t\_release

1. Purpose: Get Team List

Client: GetTeamLst

Input:

Output:

Result Whether there is an error

TeamList All Teams

Server: GET /api/team

Route: app.get (“api/team”)

Controller: SystemController/GetTeamLst

Model: Config/get

DAO: dbhelper/query

DB: t\_config

1. Purpose: Get Project List

Client: GetProjectLst

Input:

Output:

Result Whether there is an error

Release Files Download Release Files

Server: GET /api/project

Route: app.get (“api/project”)

Controller: Controller/GetProjectLst

Model: Config/get

DAO: dbhelper/query

DB: t\_config

1. Purpose: Get Work No List

Client: GetWorknoLst

Input:

Output:

Result Whether there is an error

WorknoList All Projects

Server: GET /api/user

Route: app.get (“api/user”)

Controller: UserController/GetUserLst

Model: User/get

DAO: dbhelper/query

DB: t\_user

1. Purpose: Get Customer List

Client: GetCustomerLst

Input:

ID Identify a release

Output:

Result Whether there is an error

CustomerList All Projects

Server: GET /api/customer

Route: app.get (“api/customer”)

Controller: SystemController/GetCustomerLst

Model: Config/get

DAO: dbhelper/query

DB: t\_config

1. Purpose: Get Status List

Client: GetStatusLst

Input:

Output:

Result Whether there is an error

StatusList All Status

Server: GET /api/status

Route: app.get (“api/status”)

Controller: SystemController/GetStatusLst

Model: Config/get

DAO: dbhelper/query

DB: t\_config

1. Purpose: Export excel

Client: ExportExcel

Input:

Team Identify a team

Work\_No Identify a user

Project Identify a project

Customer Identify a customer

Output:

Result Whether there is an error

Report Excel file containing release list

Server: POST /api/report

Route: app.post(“api/report”)

Controller: ResourceController/ExportExcel

Model: report/save

1. Purpose: Download Release Package

Client: Download

Input:

ID Identify a release

Output:

Result Whether there is an error

Package Release Package downloaded

Server: GET /api/package/:id

Route: app.post(“api/package”)

Controller: ResourceController/DownloadPackage

Model: package/get

### My Application Sub Page

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 我的申请 | | | | | | | | | | | |
| 项目 客户 | | | | | | | | | | 查询 | |
| 课 | 工号 | 姓名 | 项目 | 版本 | 客户 | 注释 | Svn | 申请时间 | 核准时间 | 状态 | 位置 |
| bmc | \* | \* | test | 1.0 | test | \* | \* | \* | \* | 核准 | \* |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| 首页 下一页 上一页 尾页 | | | | | | | | | | | |
| 新增 中止 | | | | | | | | | | | |

In this page, user can：

1. View all his application.
2. Query his releases by project/customer.
3. Upload a release package when approved.

Note : The page initially shows all user’s application.

Internal Interfaces:

1. Purpose: Upload Release Files

Client: Upload

Input:

ID Identify a release

Package Release Package to upload

Output:

Result Whether there is an error

Path Path of release package

Server**:** POST /api/package

Route: app.post(“/api/package”)

Controller: ResourceController/UploadPackage

Model: Package/save

DAO: dbhelper.query

DB: t\_release

### Apply Release Dialog

申请发布

项目:

版本:

客户:

Svn:

返回

确定

In this page, user can select/Input project/version/customer/svn/release notes and apply a release.

Internal Interfaces:

1. Purpose: Apply a release

Client: StartApply

Input:

Params Project/Version/Customer/Svn

Output:

Result Whether there is an error

Server**:** POST /api/release

Route: app.post(“/api/release”)

Controller: ReleaseController/StartApply

Model: Release/save

DAO: dbhelper.query

DB: t\_release

### Stop Release Dialog

中止发布

理由

返回

确定

In this page, user can input reason and stop a release.

Internal Interfaces:

1. Purpose: Stop a release

Client: StopApply

Input:

ID Identify a release

Output:

Result Whether there is an error

Server**:** DELETE /api/release/:id

Route: app.delete(“/api/apply”)

Controller: ReleaseController/StopApply

Model: Release/save

DAO: dbhelper.query

DB: t\_release

### My Sign Page ( reviewer/approver )

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 我的签核 | | | | | | | | | | | |
| 课 工号 项目 客户 申请时间 | | | | | | | | | | 查询 | |
| 课 | 工号 | 姓名 | 项目 | 版本 | 客户 | 注释 | svn | 申请时间 | 核准时间 | 状态 | 位置 |
| bmc | \* | \* | test | 1.0 | test | \* | \* | \* | \* | 核准 | \* |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| 首页 下一页 上一页 尾页 | | | | | | | | | | | |
| 签核 | | | | | | | | | | | |

In this page, reviewer/approver can select a release to sign.

### Sign Dialog

审核确认

意见：

拒绝

同意

In this dialog, reviewer/approver can approve a release or reject it.

Note: Reviewer/approver must input reason when reject a release.

Internal Interfaces:

1. Purpose: Approve/Reject a release

Client: Sign

Input:

ID Identify a release

Reason Reason to reject a release

Output:

Result Whether there is an error

Server**:** POST /api/sign/:id

Route: app.post(“/api/release”)

Controller: ReleaseController/Sign

Model: release/save

DAO: dbhelper.query

DB: t\_release

### Home Page（Administrator）

|  |  |  |
| --- | --- | --- |
| 软体发布系统 | | 工号 姓名 登出 |
| 用户管理 |  | |
| 组织管理 |  | |
| 项目管理 |  | |
| 客户管理 |
| 系统设定 |
| Copyright 2015 Foxconn CABG ©. All Rights Reserved. | | |

In this page, administrator can select a sub page and accomplish specific configuration tasks.

### User Management Sub Page

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用户管理 | | | | |
| 课 工号 | | | | 查询 |
| 课 | 工号 | 姓名 | 角色 | 备注 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 首页 下一页 上一页 尾页 | | | | |
| 新增 修改 重置 删除 | | | | |

In this page, administrator can:

1. View all users in the system.
2. Query users by team/work\_no.
3. Add a user.
4. Modify a user.
5. Reset user’s password.
6. Delete a user.

Internal interfaces:

### Add User Dialog

新增用户

课:

:

工号:

姓名:

角色:

备注:

取消

确定

In this dialog, administrator can input type/team/work\_no/name/remark and add a user.

Note: User can have several roles.

1. Purpose: Add a user

Client: AddUser

Input:

Params type/team/work\_no/name/remark

Output:

Result Whether there is an error

Server**:** POST /api/user

Route: app.post(“/api/user”)

Controller: UserController/AddUser

Model: User/save

DAO: dbhelper.query

DB: t\_user

### Modify User Dialog

修改用户

课:

:

工号:

姓名:

角色:

备注:

取消

确定

In this dialog, administrator can input roles/team/name/remark and modify a user(work no is readonly for a new work no leads to a new user).

Note: User can have several roles.

1. Purpose: Modify a user

Client: ModifyUser

Input:

Params type/team/name/remark

Output:

Result Whether there is an error

Server**:** POST /api/user/:id

Route: app.post(“/api/user”)

Controller: UserController/ModifyUser

Model: User/save

DAO: dbhelper.query

DB: t\_user

### Team Management Sub Page

|  |  |  |
| --- | --- | --- |
| 组织管理 | | |
| 名称 | | 查询 |
| 名称 | 备注 | |
|  |  | |
|  |  | |
|  |  | |
|  |  | |
| 首页 下一页 上一页 尾页 | | |
| 新增 删除 | | |

In this page, administrator can:

1. View all teams in the system.
2. Query teams by name.
3. Add a team.
4. Delete a team

### Add Team Dialog

新增组

名称:

:

备注:

取消

确定

In this dialog, administrator can input name/remark and add a team.

1. Purpose: Add a team

Client: AddTeam

Input:

Name Team name

Remark Remark of Team

Output:

Result Whether there is an error

Server**:** POST /api/team

Route: app.post(“/api/team”)

Controller: SystemController/AddTeam

Model: Config/save

DAO: dbhelper.query

DB: t\_config

### Project Management Sub Page

|  |  |  |
| --- | --- | --- |
| 项目管理 | | |
| 名称 | | 查询 |
| 名称 | 备注 | |
|  |  | |
|  |  | |
|  |  | |
|  |  | |
| 首页 下一页 上一页 尾页 | | |
| 新增 删除 | | |

In this page, administrator can:

1. View all projects in the system.
2. Query projects by name.
3. Add a project.
4. Delete a project.

### Add Project Dialog

新增项目

名称:

:

备注:

取消

确定

In this dialog, administrator can input name/remark and add a project.

1. Purpose: Add a project

Client: AddProject

Input:

Name Project name

Remark Remark of Project

Output:

Result Whether there is an error

Server**:** POST /api/project

Route: app.post(“/api/project”)

Controller: SystemController/AddProject

Model: Config/save

DAO: dbhelper.query

DB: t\_config

### Customer Management Sub Page

|  |  |  |
| --- | --- | --- |
| 客户管理 | | |
| 名称 | | 查询 |
| 名称 | 备注 | |
|  |  | |
|  |  | |
|  |  | |
|  |  | |
| 首页 下一页 上一页 尾页 | | |
| 新增 删除 | | |

In this page, administrator can:

1. View all customers in the system.
2. Query customers by name.
3. Add a customer.
4. Delete a customer.

### Add Customer Dialog

新增客户

名称:

:

备注:

取消

确定

In this dialog, administrator can input name/remark and add a customer.

1. Purpose: Add a customer

Client: AddCustomer

Input:

Name Customer name

Remark Remark of Customer

Output:

Result Whether there is an error

Server**:** POST /api/customer

Route: app.post(“/api/customer”)

Controller: SystemController/AddCustomer

Model: Config/save

DAO: dbhelper.query

DB: t\_config

## Internal Interface

1. Other Design Considerations and IMPACT

N/A

1. OPEN ISSUE and EFFORT

N/A