개요

**아래의 기본 요건에서 필수적인 부분에 대한 논의를 거쳐 새로운 공통 정의 항목을 도출해 냄**

**기존 공통 정의 항목**

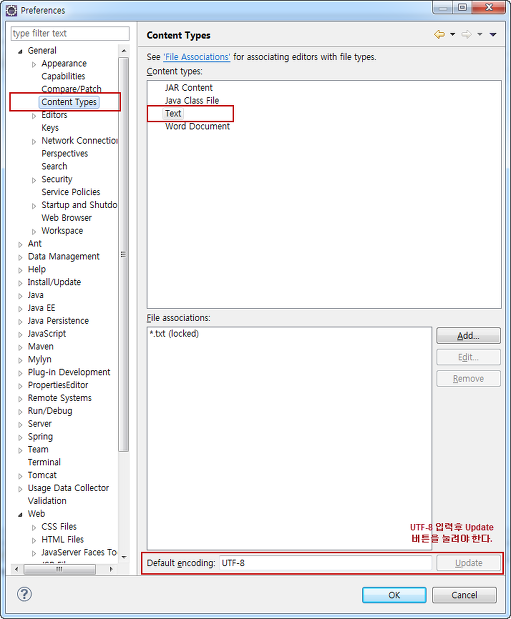
|  |  |  |  |
| --- | --- | --- | --- |
| Level 1 | Level 2 | Level 3 | Description |
| 세션관리 | 로그인 정보 | 메서드 정의 | 로그인/로그아웃에 대한 각각의 메서드 정의와 사용법 제공 |
|  |  | 로그인 유지시간 | 기본 유지시간에 대한 정의 |
|  | 공통 코드(상수) | 날짜포멧 | 공용으로 사용될 날짜포멧의 정의 |
|  |  | 문자열 셋 | 공통 문자열 셋 정의(ex: UTF-8) |
|  | 프로그램(화면) 정보 | 메뉴 구성 | 메뉴의 사용방법 제공(Session의 메뉴 정보 상태에 따라 메뉴는 동적으로 변경) |
|  | 권한 정보 |  | 권한에 대한 정의 |
| 송수신객체정의 | 로그인 | 로그인 정보 정의 |  |
|  | 메뉴정보 구성 |  | 화면에 출력되는 메뉴(사이드메뉴)의 값들로 hierarchy구조를 가질 수 있게 json 형태로 구성 |
|  | Binding Data(Grid/Tree/List) 정의 | 요건정보(Grid) | 요건정보 화면의 Grid에 바인딩되는 Data구조에 대한 정의 |
|  |  | … |  |
|  |  |  |  |
| 레이아웃 | 레이아웃 구조 정의(Header, Content, Side, Footer) |  | 화면의 형태에 대한 정의와 Template파일 제공 |
| 설정 관리 | 기본 파라메터 | 기본 메뉴 | 메인 화면의 초기 상태 정의와 Guest(비로그인)상태에서의 각 화면의 메뉴에 대한 정의 |
|  |  | 기본 컨트롤 상태 | Collapse/Expand, Enable/Disable 따위가 적용되는 컨트롤들에 대한 기본 상태 정의 |
|  | 프로그램/메뉴 정보 리소스 관리 |  | 프로그램(화면)과 메뉴의 정보를 다룰 수 있는 규칙 제공(ex: |
| 공통 스크립트 | 날짜 포멧 변경(Long<->DateTimeString) |  | YYYY/MM/DD mm:hh:ss.SSS |
|  | Restful 서비스 호출 메서드 | CRUD 기본형 정의 | ajax로 CRUD를 쉬이 할 수 있도록 Wrapper 함수를 제공 |
|  |  | Login | 로그인 메서드를 제공 |
|  |  | 세션 정보 접근 메서드 | 세션정보를 요청에 대한 함수를 미리 정의하여 제공 |
|  |  | 화면이동 | 화면 이동에 대한(URL요청 및 리다이렉션)에 대한 몇 가지 케이스를 정의하고 케이스 별 함수를 제공 |
|  | 엑셀 출력 |  | 엑셀 다운로드 기능의 제공(컨트롤의 Json 데이터 혹은 Text를 대상으로 한다) |
|  | 클립보드 | 복사 | Grid의 경우 바인딩된 Data의 Json을 복사할 수 있도록 기능을 제공한다. |
|  |  | 붙여넣기 | Json 형태의 문자열로 붙여넣기를 할 수 있도록 기능을 제공한다. Boolean 값의 경우 true/false 소문자로 반환 |
|  | UI Wrapper | DataGrid | Grid의 기본 사용과 예 제공 |
|  |  | Search | Search 기능의 일반화 및 메서드 정의 |
|  |  | ComboBox(자동완성) | 자동 완성 기능의 일반화 및 메서드 정의 |
|  |  | Editor | Editor의 저장과 로드에 대한 기능 제공 |
|  |  | Tab | 탭 기본 기능 예제 제공 |
|  |  | Dialog(modal) | Show, Hide, Terminate에 대한 기능 제공 |
|  |  | TreeView | Textbox와 연계되어 동작하는 필터링 기능 제공 |
|  |  | ListView | Multi select 기능과 추가 삭제에 대한 메서드 제공 |
| 규칙 정의 | 파일(폴더)구조 |  | 프로젝트 파일구조 정의 |
|  | 네이밍 규칙 | 변수 |  |
|  |  | 파일 |  |
|  |  | 폴더 |  |
|  | HTML 구조 정의 | 파일 공통 구조 |  |
|  |  | Header | 헤더에 정의되는 요소 |
|  |  | Content(article) |  |
|  |  | Footer |  |
|  |  | Script | 공통 스크립트 파일의 위치와 import에 대한 규칙 기술 |
|  |  | CSS | 스타일 시트 파일의 위치와 import에 대한 규칙 기술 |

**새 공통 정의 항목 리스트**

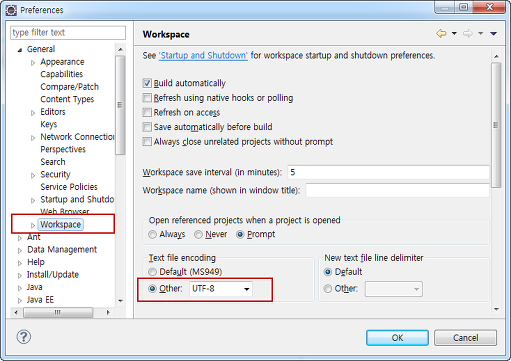
1. 메서드 정의
   1. CRUD Wrapping
      1. ajax요청과 일반 페이지 요청에 대한 일반화된 메서드를 만들고 이에 대한 문서를 제공(ex: function getRead(url,args[]) 와 같은 형태로 구성)
   2. 자주 사용되는 공통 기능에 대한 명세
      1. 일반 기능(로그인/로그아웃 등)
      2. 세션에서 관리되는 정보에 대한 정의(데이터 정의가 전제)
2. 문자열 셋 정의
   1. 시스템에 따라 쉽게 설정 할 수 있도록 가이드 제공
3. 데이터 구조
   1. 사이드 메뉴
   2. 요건조회의 그리드 출력용 데이터 조회 조건 데이터 정의
4. 템플릿 파일 구성
   1. 화면 설계 - 실제 적용될 테마에 맞는 화면 구성에 대한 설계문서
   2. header/side/footer/content 4개 파일 구성
5. 컨트롤 Wrapping
   1. grid/list/search/auto complete/tab/dialog/combo 등 구성
      1. show/hide, enable/disable, collapsable 등의 속성을 미리 정의하여 문서화
      2. copy/paste 기능 구현(일부 컨트롤에 한 함)
6. 파일 구조
   1. 소스(프로젝트)의 파일구조
   2. 페이지 소스내의 스크립트/스타일 등 위치에 대한 규칙 정의

공통 정의

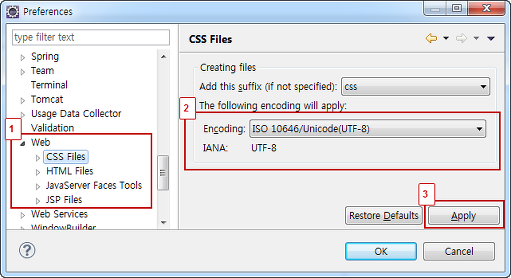
1. 메서드 정의(박민화)
   1. CRUD Wrapping
      1. ajax요청과 일반 페이지 요청에 대한 일반화된 메서드를 만들고 이에 대한 문서를 제공(ex: function getRead(url,args[]) 와 같은 형태로 구성)
   2. 자주 사용되는 공통 기능에 대한 명세
      1. 일반 기능(로그인/로그아웃 등)
      2. 세션에서 관리되는 정보에 대한 정의(데이터 정의가 전제)
2. 문자열 셋 정의(김유식)
   1. Text Content Type 설정



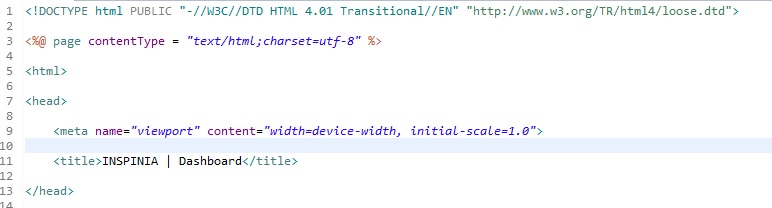
* 1. Workspace Text file Encoding 설정



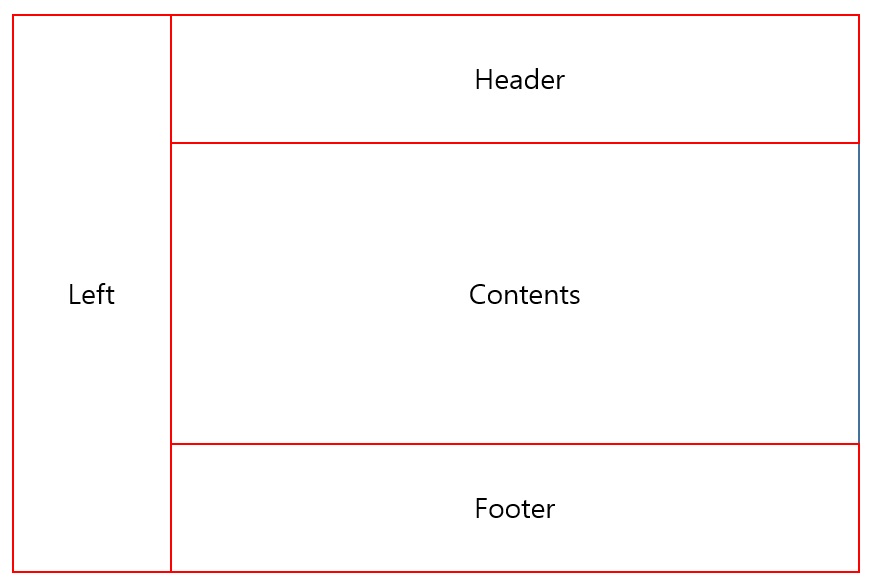
* 1. Web file encoding 설정



* 1. JSP 파일 encoding 설정



1. 데이터 구조(박민화)
   1. 사이드 메뉴
   2. 요건조회의 그리드 출력용 데이터 조회 조건 데이터 정의
2. 템플릿 파일 구성
   1. 화면 설계 - 실제 적용될 테마에 맞는 화면 목업(오혜실)
   2. header/side/footer/content 4개 파일 구성(김유식)
      1. JSP 파일 구조



1. 컨트롤 Wrapping(송민권)
   1. grid/list/search/auto complete/tab/dialog/combo 등 구성
      1. show/hide, enable/disable, collapsable 등의 속성을 미리 정의하여 문서화
      2. copy/paste 기능 구현(일부 컨트롤에 한 함)
2. 파일 구조(김유식)
   1. 소스(프로젝트)의 파일구조



* 1. 페이지 소스내의 스크립트/스타일 등 위치에 대한 규칙 정의

현재 W3C 권고안에 따르면 CSS의 위치는 <head></head>사이에 script의 위치는 </body>태그 위쪽에 포함시키도록 되어있습니다.

따라서 권고안대로 CSS는 <head>태그 사이에, script의 위치는 </body>태그 위쪽에 위치하도록 하겠습니다.

개발 환경 구성

# # 프로젝트 압축 파일로 새로 구성

1. STS(Spring tool suite) 설치

2. Workspace 폴더 구성 및 프로젝트 파일(workspace.zip) 압축 해제

3. repositorySetting.xml 설정

- maven 을 통해 다운로드 받을 라이브러리들에 대한 정보가 있는 파일.

- localRepository의 경로를 설정하는 것으로 동봉된 repository 폴더의 위치로 지정하면 별도의 추가 다운로드 없이 사용할 수 있다.

- 파일의 경로는 프로젝트 루트를 기준으로 "/src/main/webapp/WEB-INF/repositorySetting.xml”

4. Eclipse 에서 다음으로 이동한다.(maven 설치가 전제된다. STS의 경우 maven이 포함되어져 있다.)

"Windows -> Preference->User Setting"

User Settings의 경로에 수정된 repositorySetting.xml 파일을 지정하고 Update Settings 버튼을 클릭한다.

repository 폴더가 지정되어져 있고 내부에 파일이 존재하면 잠시 후 작업이 완료되고

없다면 다운로드를 수행한다.

5. 서버창(없을 경우 Windows->perspective->servers)에서 우클릭 후 서버 생성

- 로컬의 Tomcat 경로를 지정하여 서버를 생성

- 리소스 등록에서 IPS 프로젝트를 Configured로 이동

- 생성된 서버를 start 시키고 화면을 확인

# #SVN으로 받기

1. STS 설치

2. subversion 설치

3. https://defect.mocomsys.com/svn/iip/IPS\_Client 추가

4. 다운로드

5. #프로젝트 압축파일 구성의 3 ~ 5번 수행

# \* JavaScript validation error

- Javascript 특히 jquery.min.js의 경우 eclipse의 javascript validator에서 지속적으로 에러를 표시한다.

http://blog.daum.net/wonju114/95 를 참조하여 validator에서 제외하면 된다.