

Présentation

L’application doit

- afficher la liste des salles (n° salle, libellé salle, étage, code et nom du département, nombre d’équipements),
- à partir de la liste des salles, afficher, modifier, supprimer la salle sélectionnée ou encore ajouter une nouvelle salle

L’édition d’une salle affiche les informations

- numéro, libellé, étage
- code, nom, responsable du département de la salle
- la liste modifiable des types d’équipements (libellé, nb) contenus dans la salle

La saisie du code du département doit afficher en temps réel le nom du département et son responsable.

Le numéro d’une salle est uniquement composé de caractères alphabétiques majuscules ou minuscules et de chiffres.

Le travail va consister à développer une application de bureau web sur la gestion de l’inventaire des types d’équipement des salles.

Le chef de projet a fait l’étude conceptuelle des données et des traitements.

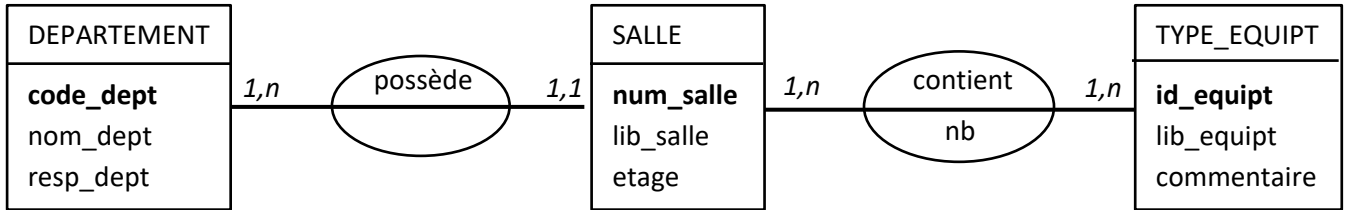
Il remet à l’équipe de développement :

- ① l’extrait du schéma conceptuel des données,
- ② le schéma relationnel de la base de données,
- ③ le script SQL d’implémentation de la base de données,
- ④ le modèle des tâches,
- ⑤ les maquettes de l’interface.

① schéma conceptuel des données de la base « bدينventaire »

de l'utilisateur « toto3u » (identifiant : « toto3u_appli », mot de passe : « 1456378 »)

l'utilisateur toto3u et ses codes d'accès sont purement fictifs.



② Schéma relationnel

DEPARTEMENT (**code_dept**, nom_dept, resp_dept)

SALLE (**num_salle**, lib_salle, etage, code_dept) tel que code_dept : clé étrangère

TYPE_EQUIPT (**id_equip**, lib_equip, commentaire)

contient (**num_salle**, **id_equip**, nb) tels que num_salle : clé étrangère ; id_equip : clé étrangère

③ le script SQL d'implémentation de la base de données

```

SET FOREIGN_KEY_CHECKS=0;

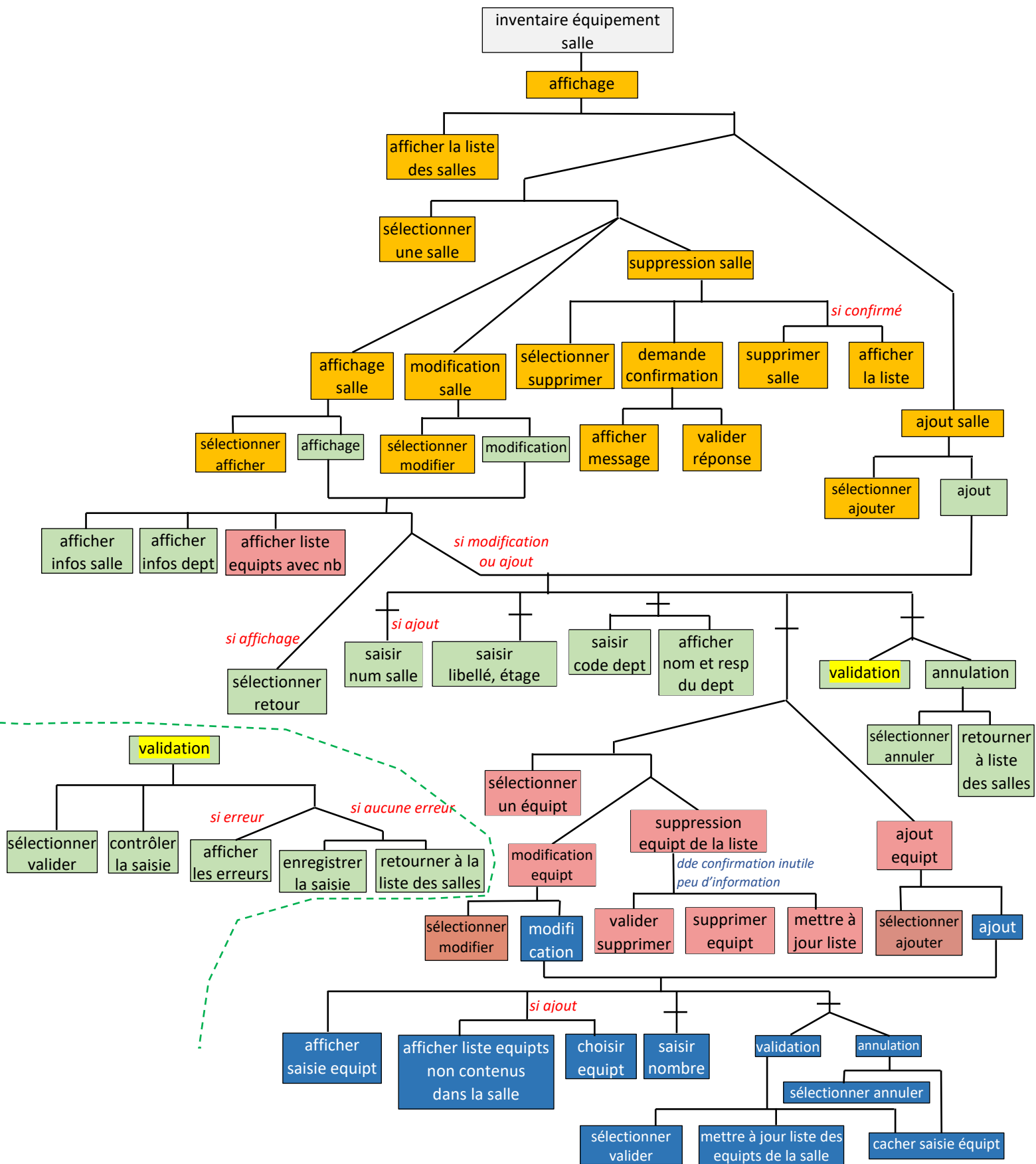
DROP TABLE IF EXISTS `contient`;
CREATE TABLE `contient` (
  `num_salle` varchar(5) NOT NULL,
  `id_equip` int(11) NOT NULL,
  `qte` int(11) NOT NULL,
  PRIMARY KEY (`num_salle`,`id_equip`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `departement`;
CREATE TABLE `departement` (
  `code_dept` varchar(5) NOT NULL,
  `nom_dept` varchar(50) NOT NULL,
  `resp_dept` varchar(30) NOT NULL,
  PRIMARY KEY (`code_dept`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `salle`;
CREATE TABLE `salle` (
  `num_salle` varchar(5) NOT NULL,
  `lib_salle` varchar(30) DEFAULT NULL,
  `etage` varchar(5) NOT NULL,
  `code_dept` varchar(30) NOT NULL,
  PRIMARY KEY (`num_salle`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `type_equip`;
CREATE TABLE `type_equip` (
  `id_equip` int(11) NOT NULL,
  `lib_equip` varchar(30) NOT NULL,
  `commentaire` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id_equip`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
    
```

④ le modèle des tâches



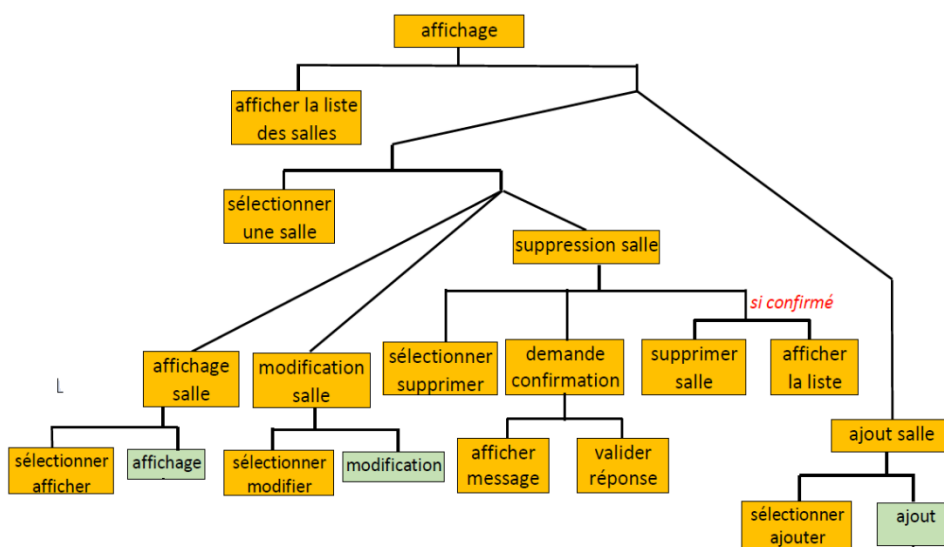
⑤ les maquettes de l’interface

Un des critères d’un bon développement est la modularité d’une application. La modularité permet de développer et de tester son interface et le code associé, de manière indépendante des autres interfaces.

Le modèle des tâches fait ressortir les interfaces possibles auxquelles on va donner un nom pour chacune. Ce nommage servira par la suite et sera utilisé pour le nommage des fichiers « html », « css » et « ts/js » associés à chaque interface.

la liste des salles

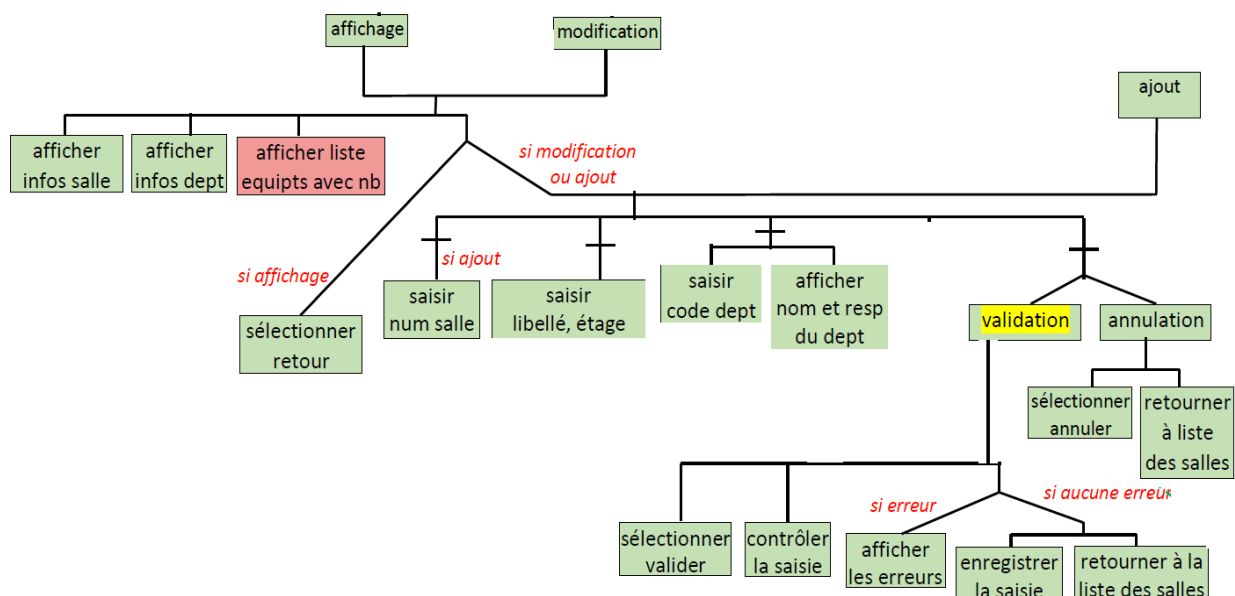
salle_liste



	Numéro	Libellé	Etage	Département	Equipements		
👁	B16	Labo de langue	1	GEA	Gestion des Entreprises et Administrations	16	🔍 🗑
👁	F13	machine PC	1	INFO	Informatique	16	🔍 🗑
👁	E23		2	INFO	Informatique	1	🔍 🗑
👁	C14		1	GEA	Gestion des Entreprises et Administrations	33	🔍 🗑
👁	E36		3	INFO	Informatique	2	🔍 🗑
👁	F39	machine PC	3	INFO	Informatique	62	🔍 🗑
							Ajouter

l'édition des informations d'une salle

salle_edit



Détail/Nouvelle/Modification salle

IDENTIFICATION

numéro

étage

DEPARTEMENT

code

 Informatique
Responsable : M. FLAGORETTE

EQUIPEMENTS

Nombre total d'équipements : 62

libellé	quantité		
chaise	16		
bureau	1		
table double	12		
vidéo-projecteur	1		
souris	16		
clavier	16		

Choisir un équipement dans la liste

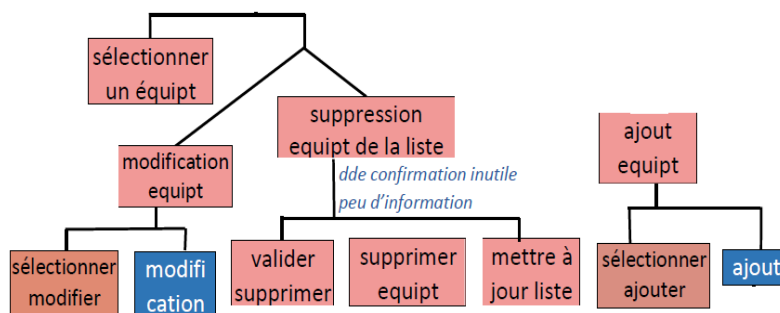
table simple	quantité <input type="text"/>
tableau VP	
tableau blanc	
tableau craie	
ordinateur	
téléviseur	

auteur : P. NITSCHKE

IHM – TypeScript - Application dirigée

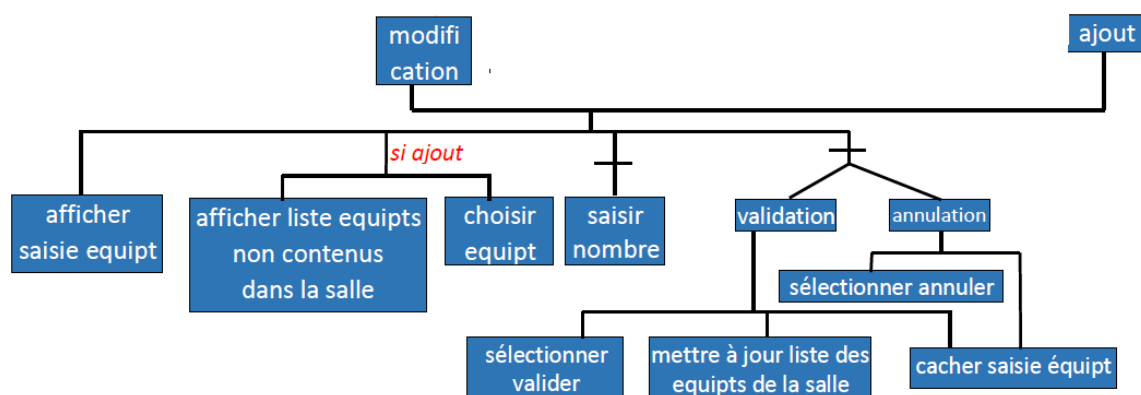
5

la liste des équipements d’une salle



Nombre total d'équipements : 62			
libellé	quantité		
chaise	16		
bureau	1		
table double	12		
vidéo-projecteur	1		
souris	16		
clavier	16		
			Ajouter

l'édition d'un équipement d'une salle



Choisir un équipement dans la liste

table simple	<div>quantité <input type="text"/></div> <div> </div>
tableau VP	
tableau blanc	
tableau craie	
ordinateur	
téléviseur	