

**PITS**

**DOCUMENTATION TECHNIQUE**

**VERSION**  
1.0

**AUTEUR**  
ALAIN POMIROL

Date de dernière mise à jour : mercredi 4 août 2010

# Table des matières

<b>1 - Historique du document.....</b>	<b>5</b>
<b>2 - Description générale de l'application.....</b>	<b>6</b>
2.1 - Présentation générale.....	6
2.2 - Base de données.....	6
<b>2.2.1 - Schéma de la base.....</b>	<b>6</b>
<b>2.2.2 - Description des tables :.....</b>	<b>6</b>
2.2.2.1 - table classes.....	6
2.2.2.2 - table eleves.....	7
2.2.2.3 - table etablissements.....	8
2.2.2.4 - table rythmesdepaiement.....	8
2.2.2.5 - table stations.....	8
2.2.2.6 - table tarifs.....	8
2.2.2.7 - table user.....	9
2.2.2.8 - table villes.....	9
<b>2.2.3 - Description des vues.....</b>	<b>10</b>
2.2.3.1 - vue elevelist.....	10
2.2.3.2 - vue elevesr1.....	11
2.2.3.3 - vue stationsvilles.....	12
2.3 - Arborescence de l'application.....	13
2.4 - Fichiers de configuration.....	14
<b>2.4.1 - Description des fichiers ini.....</b>	<b>14</b>
2.4.1.1 - Fichier config.ini.....	14
2.4.1.2 - Fichier menu.ini.....	15
2.4.1.3 - Fichier session.ini.....	16
<b>2.4.2 - Description des fichiers txt.....</b>	<b>17</b>
2.4.2.1 - Fichier mail_eleve.txt.....	17
2.4.2.2 - Fichier mail_user_debloque.txt.....	17
2.4.2.3 - Fichier mail_user_modif.txt.....	18
2.4.2.4 - Fichier mail_user_premier.txt.....	18
2.4.2.5 - pdf_eleve.txt.....	19
<b>2.4.3 - Langage pour fichier txt.....</b>	<b>21</b>
2.4.3.1 - Éléments du langage.....	21
2.4.3.1.1 - Les champs de tables et des vues de la base de donnée.....	21
2.4.3.1.2 - Syntaxe.....	21
2.4.3.2 - Interpréteur et classes dérivées.....	22
2.4.3.2.1 - Pits_Document_Structure.....	22
2.4.3.2.2 - Les classes dérivées : PdfEleve et UserMailContent.....	22
2.5 - Fichiers de logs.....	23
2.6 - Fichiers de documentation.....	23
<b>3 - Organisation de l'application.....</b>	<b>24</b>

3.1 - Les vues de l'application.....	24
<b>3.1.1 - Organisation, images et logo.....</b>	<b>24</b>
<b>3.1.2 - Statistiques de fréquentation des pages.....</b>	<b>24</b>
3.2 - Les controllers.....	24
<b>3.2.1 - Accès anonyme : IndexController.php.....</b>	<b>24</b>
3.2.1.1 - Page d'accueil - indexAction .....	24
3.2.1.2 - Création d'un compte - creationAction.....	24
3.2.1.3 - Aide - helpAction.....	25
3.2.1.4 - Envoi d'un mot de passe - passwdAction.....	25
<b>3.2.2 - Accès parent : ParentController.php.....</b>	<b>25</b>
3.2.2.1 - Page d'accueil - indexAction.....	25
3.2.2.2 - Inscription d'un enfant - inscriptioneleveAction.....	25
3.2.2.3 - Modification d'une inscription - editeleveAction.....	25
3.2.2.4 - Supprime un enfant - suppreleveAction.....	26
3.2.2.5 - Envoie un pdf de la fiche élève par mail - maileleveAction.....	26
3.2.2.6 - Modifier son compte - modifcompteAction.....	26
3.2.2.7 - Aide - helpAction.....	26
3.2.2.8 - Déconnexion - logoutAction.....	26
<b>3.2.3 - Accès administrateur : AdminController.php.....</b>	<b>26</b>
3.2.3.1 - Page d'accueil - indexAction.....	26
3.2.3.2 - Affiche la liste de utilisateurs - listusersAction.....	26
3.2.3.3 - Débloque un compte - debloqueAction.....	27
3.2.3.4 - Change un mot de passe - changemdpAction.....	27
3.2.3.5 - Supprime un utilisateur - suppruserAction.....	27
3.2.3.6 - Affiche le liste des élèves - listelevesAction.....	27
3.2.3.7 - Voir la liste des enfants d'un utilisateur - voirenfantsAction.....	28
3.2.3.8 - Modification d'une fiche d'élève - editeleveAction.....	28
3.2.3.9 - Supprime un élève - suppreleveAction.....	28
3.2.3.10 - Crée un pdf de la fiche élève et l'envoie à l'écran - pdfeleveAc tion.....	28
3.2.3.11 - Page d'accès aux statistiques - statistiquesAction.....	28
3.2.3.12 - Statistiques par point d'arrêt - statparstationAction.....	28
3.2.3.13 - Statistiques par établissement - statparetabAction.....	28
3.2.3.14 - Statistiques par commune - statparvilleAction.....	29
3.2.3.15 - Extraction csv pour BdTS - extractionAction.....	29
3.2.3.16 - Confirmation des inscriptions dans BdTS - validinscriptionsAction.....	29
3.2.3.17 - Aide - helpAction.....	29
3.2.3.18 - Met en ligne un document d'aide- addhelpAction.....	30
3.2.3.19 - Supprime un document d'aide- supprhelpAction.....	30
3.2.3.20 - Déconnexion - logoutAction.....	30
<b>3.2.4 - Accès super administrateur : SadminController.php.....</b>	<b>30</b>
3.2.4.1 - Page d'accueil - indexAction.....	30
3.2.4.2 - Récupérer et vider le fichier log -razlogAction.....	30
3.2.4.3 - Gestion des administrateurs - adminigestAction.....	30
3.2.4.4 - Ajout d'un administrateur - adminaddAction.....	30
3.2.4.5 - Modification d'un administrateur - admineditAction.....	30
3.2.4.6 - Suppression d'un administrateur - adminsupprAction.....	30

3.2.4.7 - Mot de passe d'un administrateur - adminmdpAction.....	<a href="#"><u>31</u></a>
3.2.4.8 - Vider la table des élèves - razelevesAction.....	<a href="#"><u>31</u></a>
3.2.4.9 - Vider la table des utilisateurs - razuserAction.....	<a href="#"><u>31</u></a>
3.2.4.10 - Upload et exécute un fichier sql - sqlAction.....	<a href="#"><u>31</u></a>
3.2.4.11 - Décrit les tables - decritablesAction.....	<a href="#"><u>31</u></a>
3.2.4.12 - Liste des fichiers de configuration - editconfigAction.....	<a href="#"><u>31</u></a>
3.2.4.13 - Edite un fichier du répertoire de configuration- editfichierAction.....	<a href="#"><u>31</u></a>
3.2.4.14 - Change le mot de passe de sadmin - changempdpAction.....	<a href="#"><u>32</u></a>
3.2.4.15 - Déconnexion - logoutAction.....	<a href="#"><u>32</u></a>
<b>3.2.5 - Gestion du login : LoginController.php.....</b>	<a href="#"><b><u>32</u></b></a>
3.2.5.1 - indexAction.....	<a href="#"><u>32</u></a>
3.2.5.2 - Annulation de la création d'un compte - annulationAction.....	<a href="#"><u>32</u></a>
3.2.5.3 - Validation de la création d'un compte - validationAction.....	<a href="#"><u>32</u></a>
3.2.5.4 - Login - loginAction.....	<a href="#"><u>32</u></a>
3.2.5.5 - Logout - logoutAction.....	<a href="#"><u>32</u></a>
3.3 - Gestion des erreurs .....	<a href="#"><u>33</u></a>
3.3.1 - Erreurs d'Url.....	<a href="#"><u>33</u></a>
3.3.2 - Interception des erreurs du programme.....	<a href="#"><u>33</u></a>
3.3.3 - Gestion des exceptions - ErrorController.....	<a href="#"><u>33</u></a>
<b>4 - Comptes d'administrateur et de super administrateur.....</b>	<a href="#"><b><u>34</u></b></a>
4.1 - Comptes d'administrateurs.....	<a href="#"><u>34</u></a>
4.2 - Comptes de super administrateurs.....	<a href="#"><u>34</u></a>

1 - Historique du document

Date	Auteur	Version	Sujet de la modification
29/07/2010	Pomirol Alain	1.0	Création

## 2 - Description générale de l'application

### 2.1 - Présentation générale

PiTS est une application développée en Php - MySql, en utilisant Zend Framework.

Elle fonctionne sous Php (version 5.2.6 ou supérieur), apache (version 2.0 ou supérieur) et MySql (version 5.0 ou supérieur).

Elle utilise le Zend Framework de version 1.10.3.

### 2.2 - Base de données

La base de données de l'application est une base MySql version 5.0 ou supérieur.

Elle est composée de 8 tables de type MyISAM, interclassement utf8\_general\_ci, et de 3 vues.

#### 2.2.1 - Schéma de la base

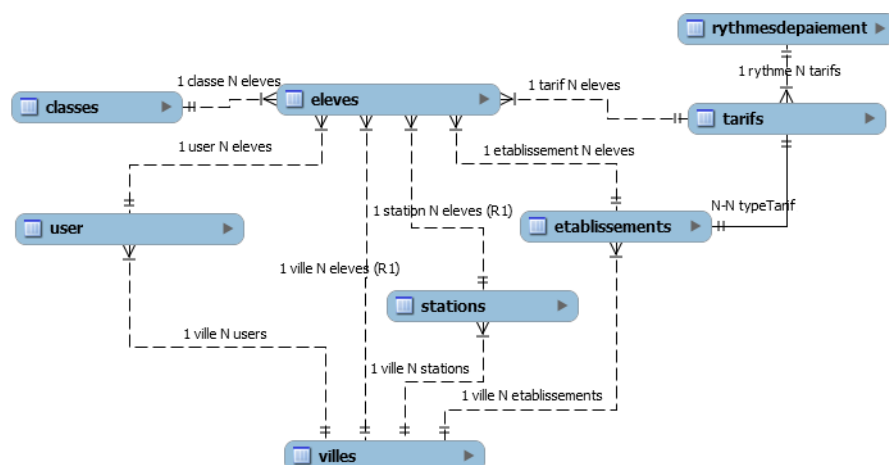


Illustration 1: Schéma de la base de données de PiTS

#### 2.2.2 - Description des tables :

##### 2.2.2.1 - table classes

Structure de la table

```

1 DROP TABLE IF EXISTS `classes`;
2 CREATE TABLE IF NOT EXISTS `classes` (
3   `CodeClasse` int(11) NOT NULL,
4   `Nom` varchar(30) NOT NULL,
5   PRIMARY KEY (`CodeClasse`)
6 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les données de cette table proviennent de BdTS.

### 2.2.2.2 - table eleves

#### Structure de la table

```

1 DROP TABLE IF EXISTS `eleves`;
2 CREATE TABLE IF NOT EXISTS `eleves` (
3   `eleveId` int(11) NOT NULL AUTO_INCREMENT,
4   `userId` int(11) NOT NULL DEFAULT '0' COMMENT 'index table user',
5   `Nom` varchar(30) NOT NULL,
6   `Prenom` varchar(30) NOT NULL,
7   `DateN` date NOT NULL COMMENT 'format YYYY-MM-DD',
8   `CodeEN` varchar(8) NOT NULL COMMENT 'index table etablissements',
9   `Regime` tinyint(1) NOT NULL DEFAULT '1',
10  `Classe` int(11) NOT NULL DEFAULT '0' COMMENT 'index table classes',
11  `TitreR1` varchar(7) NOT NULL,
12  `NomR1` varchar(30) NOT NULL,
13  `PrenomR1` varchar(30) NOT NULL,
14  `AdressR1L1` varchar(30) NOT NULL,
15  `AdressR1L2` varchar(30) DEFAULT NULL,
16  `CommuneR1` varchar(5) NOT NULL COMMENT 'index table villes',
17  `CodePostalR1` varchar(5) NOT NULL,
18  `TelephoneR1` varchar(10) DEFAULT NULL,
19  `TelephoneR1c` varchar(10) DEFAULT NULL,
20  `Email` varchar(80) DEFAULT NULL,
21  `CodeStationR1` int(11) NOT NULL DEFAULT '0' COMMENT 'index table stations',
22  `CodeTarif` int(11) NOT NULL DEFAULT '0' COMMENT 'index table tarifs',
23  `RibBanque` varchar(5) NOT NULL,
24  `RibAgence` varchar(5) NOT NULL,
25  `RibCompte` varchar(11) NOT NULL,
26  `RibCle` varchar(2) NOT NULL,
27  `RibDom` varchar(24) NOT NULL,
28  `RibTit` varchar(24) NOT NULL,
29  `SecondeAdresse` tinyint(1) NOT NULL DEFAULT '0',
30  `TitreR2` varchar(7) DEFAULT NULL,
31  `NomR2` varchar(30) DEFAULT NULL,
32  `PrenomR2` varchar(30) DEFAULT NULL,
33  `AdressR2L1` varchar(30) DEFAULT NULL,
34  `AdressR2L2` varchar(30) DEFAULT NULL,
35  `CommuneR2` varchar(5) DEFAULT NULL COMMENT 'null ou index table villes',
36  `CodePostalR2` varchar(5) DEFAULT NULL,
37  `TelephoneR2` varchar(10) DEFAULT NULL,
38  `TelephoneR2c` varchar(10) DEFAULT NULL,
39  `CodeStationR2` int(11) DEFAULT NULL COMMENT 'null ou index table stations',
40  `dateCreation` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
41  `dateModif` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
42  `dateExtraction` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
43  `dateValidation` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
44  `ficheExtraite` tinyint(1) NOT NULL DEFAULT '0',
45  `ficheValidee` tinyint(1) NOT NULL DEFAULT '0',
46  `ficheModifiee` tinyint(1) NOT NULL DEFAULT '0',
47  PRIMARY KEY (`eleveId`),
48  KEY `userId` (`userId`)
49 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les dates de suivi des opérations sont au format "YYYY-MM-dd HH:mm:ss".

Les données de cette table proviennent de la saisie en ligne des utilisateurs.

Cette table doit être réinitialisée chaque année, au lancement de la campagne d'inscriptions.

### 2.2.2.3 - table etablisements

#### Structure de la table

```
1 DROP TABLE IF EXISTS `etablisements`;  
2 CREATE TABLE IF NOT EXISTS `etablisements` (  
3   `CodeEN` varchar(8) NOT NULL,  
4   `Nom` varchar(30) NOT NULL,  
5   `Commune` varchar(5) NOT NULL COMMENT 'index table villes',  
6   `Niveau` tinyint(4) NOT NULL,  
7   `TypeTarif` tinyint(4) NOT NULL COMMENT 'lien Tarifs.TypeTarif',  
8   PRIMARY KEY (`CodeEN`)  
9 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les données de cette table proviennent de BdTS.

### 2.2.2.4 - table rythmesdepaieement

#### Structure de la table

```
1 DROP TABLE IF EXISTS `rythmesdepaieement`;  
2 CREATE TABLE IF NOT EXISTS `rythmesdepaieement` (  
3   `CodeRythme` int(11) NOT NULL,  
4   `Rythme` varchar(25) NOT NULL,  
5   PRIMARY KEY (`CodeRythme`)  
6 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les données de cette table proviennent de BdTS.

### 2.2.2.5 - table stations

#### Structure de la table

```
1 DROP TABLE IF EXISTS `stations`;  
2 CREATE TABLE IF NOT EXISTS `stations` (  
3   `CodeStation` int(11) NOT NULL,  
4   `Commune` varchar(5) NOT NULL COMMENT 'index table villes',  
5   `Nom` varchar(35) NOT NULL,  
6   `garder` tinyint(1) NOT NULL,  
7   PRIMARY KEY (`CodeStation`)  
8 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les données de cette table proviennent de BdTS.

### 2.2.2.6 - table tarifs

#### Structure de la table

```
1 DROP TABLE IF EXISTS `tarifs`;  
2 CREATE TABLE IF NOT EXISTS `tarifs` (  
3   `CodeTarif` int(11) NOT NULL,  
4   `Tarif` decimal(10,2) NOT NULL,  
5   `Libelle` varchar(30) NOT NULL,  
6   `CodeRythme` int(4) NOT NULL COMMENT 'index table rythmesdepaieement',  
7   `Prelevement` tinyint(1) NOT NULL,  
8   `TypeTarif` tinyint(4) NOT NULL COMMENT 'lien Etablisements.TypeTarif',  
9   PRIMARY KEY (`CodeTarif`)  
10 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```



Les données de cette table proviennent de BdTS.

### 2.2.2.7 - table user

Structure de la table

```
1 DROP TABLE IF EXISTS `user`;  
2 CREATE TABLE IF NOT EXISTS `user` (  
3   `userId` int(11) NOT NULL AUTO_INCREMENT,  
4   `Nom` varchar(30) NOT NULL,  
5   `Prenom` varchar(30) NOT NULL,  
6   `AdresseL1` varchar(30) NOT NULL,  
7   `AdresseL2` varchar(30) NOT NULL,  
8   `Commune` varchar(5) NOT NULL COMMENT 'index table villes',  
9   `CodePostal` varchar(5) NOT NULL,  
10  `Titre` varchar(8) NOT NULL,  
11  `Telephone` varchar(10) NOT NULL,  
12  `TelephoneC` varchar(10) NOT NULL,  
13  `Email` varchar(80) NOT NULL COMMENT 'login',  
14  `mdp` varchar(64) NOT NULL COMMENT 'code sha1',  
15  `categorie` tinyint(4) NOT NULL DEFAULT '1',  
16  `bloque` tinyint(1) NOT NULL DEFAULT '1',  
17  `dateCreation` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
18  `dateModif` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
19  `dateLastLogin` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
20  `lastIp` varchar(16) NOT NULL,  
21  `datePreviousLogin` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
22  `previousIp` varchar(16) NOT NULL,  
23  `temoin` varchar(32) NOT NULL,  
24  PRIMARY KEY (`userId`),  
25  UNIQUE KEY `Email` (`Email`)  
26 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les données de cette table proviennent de la saisie en ligne des utilisateurs.

Cette table doit être réinitialisée chaque année, au lancement de la campagne d'inscriptions.

### 2.2.2.8 - table villes

Structure de la table

```
1 DROP TABLE IF EXISTS `villes`;  
2 CREATE TABLE IF NOT EXISTS `villes` (  
3   `CodeInsee` varchar(5) NOT NULL,  
4   `Nom` varchar(30) NOT NULL,  
5   `CodePostal` varchar(5) NOT NULL,  
6   `Adhesion` tinyint(1) NOT NULL COMMENT 'Membre de la CCM',  
7   PRIMARY KEY (`CodeInsee`),  
8   KEY `Nom` (`Nom`)  
9 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Les données de cette table proviennent de BdTS.

Dans les listes de choix, seules les villes dont le champ *Adhesion* est 1 seront proposées.

## 2.2.3 - Description des vues

### 2.2.3.1 - vue elevelist

#### Déclaration de la vue

```

1 CREATE VIEW `eleveslist` AS
2 SELECT `elevesr1`.`eleveId` AS `eleveId`,
3        `elevesr1`.`userId` AS `userId`,
4        `elevesr1`.`Nom` AS `Nom`,
5        `elevesr1`.`Prenom` AS `Prenom`,
6        `elevesr1`.`DateN` AS `DateN`,
7        `elevesr1`.`VilleEcole` AS `VilleEcole`,
8        `elevesr1`.`NomEcole` AS `NomEcole`,
9        `elevesr1`.`Regime` AS `Regime`,
10       `elevesr1`.`Classe` AS `Classe`,
11       `elevesr1`.`TitreR1` AS `TitreR1`,
12       `elevesr1`.`NomR1` AS `NomR1`,
13       `elevesr1`.`PrenomR1` AS `PrenomR1`,
14       `elevesr1`.`AdressR1L1` AS `AdressR1L1`,
15       `elevesr1`.`AdressR1L2` AS `AdressR1L2`,
16       `elevesr1`.`NomVilleR1` AS `NomVilleR1`,
17       `elevesr1`.`CodePostalR1` AS `CodePostalR1`,
18       `elevesr1`.`TelephoneR1` AS `TelephoneR1`,
19       `elevesr1`.`TelephoneR1c` AS `TelephoneR1c`,
20       `elevesr1`.`Email` AS `Email`,
21       `elevesr1`.`VilleStationR1` AS `VilleStationR1`,
22       `elevesr1`.`NomStationR1` AS `NomStationR1`,
23       `elevesr1`.`TarifLibelle` AS `TarifLibelle`,
24       `elevesr1`.`TarifTarif` AS `TarifTarif`,
25       `elevesr1`.`TarifPrelevement` AS `TarifPrelevement`,
26       `elevesr1`.`RibBanque` AS `RibBanque`,
27       `elevesr1`.`RibAgence` AS `RibAgence`,
28       `elevesr1`.`RibCompte` AS `RibCompte`,
29       `elevesr1`.`RibCle` AS `RibCle`,
30       `elevesr1`.`RibDom` AS `RibDom`,
31       `elevesr1`.`RibTit` AS `RibTit`,
32       `elevesr1`.`SecondeAdresse` AS `SecondeAdresse`,
33       `elevesr1`.`TitreR2` AS `TitreR2`,
34       `elevesr1`.`NomR2` AS `NomR2`,
35       `elevesr1`.`PrenomR2` AS `PrenomR2`,
36       `elevesr1`.`AdressR2L1` AS `AdressR2L1`,
37       `elevesr1`.`AdressR2L2` AS `AdressR2L2`,
38       `villes`.`Nom` AS `NomVilleR2`,
39       `elevesr1`.`CodePostalR2` AS `CodePostalR2`,
40       `elevesr1`.`TelephoneR2` AS `TelephoneR2`,
41       `elevesr1`.`TelephoneR2c` AS `TelephoneR2c`,
42       `villes_1`.`Nom` AS `VilleStationR2`,
43       `stations`.`Nom` AS `NomStationR2`
44 FROM (((`elevesr1` LEFT JOIN `villes` ON((`elevesr1`.`CommuneR2` =
45 `villes`.`CodeInsee`))) LEFT JOIN `stations` ON((`elevesr1`.`CodeStationR2` =
46 `stations`.`CodeStation`))) LEFT JOIN `villes` `villes_1`
47 ON((`stations`.`Commune` = `villes_1`.`CodeInsee`)));

```

Cette vue est basée sur la vue suivante (voir elevsr1)

### 2.2.3.2 - vue elevesr1

#### Déclaration de la vue

```

1 CREATE VIEW `elevesr1` AS
2 SELECT `eleves`.`eleveId` AS `eleveId`,
3        `eleves`.`userId` AS `userId`,
4        `eleves`.`Nom` AS `Nom`,
5        `eleves`.`Prenom` AS `Prenom`,
6        `eleves`.`DateN` AS `DateN`,
7        `villes_1`.`Nom` AS `VilleEcole`,
8        `etablisements`.`Nom` AS `NomEcole`,
9        `eleves`.`Regime` AS `Regime`,
10       `classes`.`Nom` AS `Classe`,
11       `eleves`.`TitreR1` AS `TitreR1`,
12       `eleves`.`NomR1` AS `NomR1`,
13       `eleves`.`PrenomR1` AS `PrenomR1`,
14       `eleves`.`AdressR1L1` AS `AdressR1L1`,
15       `eleves`.`AdressR1L2` AS `AdressR1L2`,
16       `villes`.`Nom` AS `NomVilleR1`,
17       `eleves`.`CodePostalR1` AS `CodePostalR1`,
18       `eleves`.`TelephoneR1` AS `TelephoneR1`,
19       `eleves`.`TelephoneR1c` AS `TelephoneR1c`,
20       `eleves`.`Email` AS `Email`,
21       `villes_2`.`Nom` AS `VilleStationR1`,
22       `stations`.`Nom` AS `NomStationR1`,
23       `tarifs`.`Libelle` AS `TarifLibelle`,
24       `tarifs`.`Tarif` AS `TarifTarif`,
25       `tarifs`.`Prelevement` AS `TarifPrelevement`,
26       `eleves`.`RibBanque` AS `RibBanque`,
27       `eleves`.`RibAgence` AS `RibAgence`,
28       `eleves`.`RibCompte` AS `RibCompte`,
29       `eleves`.`RibCle` AS `RibCle`,
30       `eleves`.`RibDom` AS `RibDom`,
31       `eleves`.`RibTit` AS `RibTit`,
32       `eleves`.`SecondeAdresse` AS `SecondeAdresse`,
33       `eleves`.`TitreR2` AS `TitreR2`,
34       `eleves`.`NomR2` AS `NomR2`,
35       `eleves`.`PrenomR2` AS `PrenomR2`,
36       `eleves`.`AdressR2L1` AS `AdressR2L1`,
37       `eleves`.`AdressR2L2` AS `AdressR2L2`,
38       `eleves`.`CommuneR2` AS `CommuneR2`,
39       `eleves`.`CodePostalR2` AS `CodePostalR2`,
40       `eleves`.`TelephoneR2` AS `TelephoneR2`,
41       `eleves`.`TelephoneR2c` AS `TelephoneR2c`,
42       `eleves`.`CodeStationR2` AS `CodeStationR2`
43 FROM (((`stations` JOIN (`etablisements` join (`tarifs` JOIN (`villes` JOIN
44 (`classes` join `eleves` on((`classes`.`CodeClasse` = `eleves`.`Classe`)))
45 ON((`villes`.`CodeInsee` = `eleves`.`CommuneR1`))) ON((`tarifs`.`CodeTarif` =
46 `eleves`.`CodeTarif`))) on((`etablisements`.`CodeEN` = `eleves`.`CodeEN`)))
47 ON((`stations`.`CodeStation` = `eleves`.`CodeStationR1`))) JOIN `villes`
48 `villes_2` ON((`stations`.`Commune` = `villes_2`.`CodeInsee`))) JOIN `villes`
49 `villes_1` ON((`etablisements`.`Commune` = `villes_1`.`CodeInsee`)));

```

### 2.2.3.3 - vue stationsvilles

#### Déclaration de la vue

```
1 CREATE VIEW `stationsvilles` AS
2 SELECT `stations`.`CodeStation` AS `CodeStation`,
3        `stations`.`Commune` AS `codeInsee`,
4        `villes`.`Nom` AS `NomVille`,
5        `stations`.`Nom` AS `NomStation`
6 FROM (`stations` JOIN `villes` ON((`stations`.`Commune` =
7 `villes`.`CodeInsee`)))
8 ORDER BY `stations`.`Commune`,`stations`.`Nom`;
```

## 2.3 - Arborescence de l'application

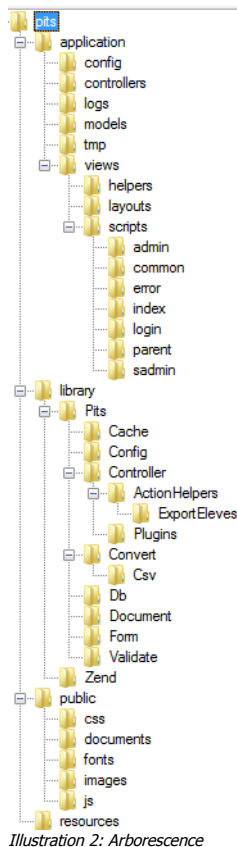


Illustration 2: Arborescence

1. Le répertoire *pits/application/config* contient
  - a. les fichiers de configuration de l'application ;
  - b. les modèles de mails et de pdf.
 Ces fichiers peuvent être gérés par l'application en *super administrateur* si le répertoire a des droits d'écriture pour apache.
2. Le répertoire *pits/application/logs* doit avoir des droits d'écriture pour apache. Il contient le fichier de logs.
3. Le répertoire *pits/application/tmp* doit avoir des droits d'écriture pour apache. Il contient les fichiers de sessions.
4. Le répertoire *pits/public/documents* doit avoir des droits d'écriture pour apache. Il contient les fichiers que vous souhaitez proposer aux utilisateurs par le menu **aide** de l'application. Vous placerez ces documents par votre compte d'administrateur.
5. La librairie Zend\_Framework est contenue dans le répertoire *pits/library/Zend*.
6. Tous les répertoires et fichiers de l'arborescence sont accessibles en lecture par apache.
7. Le répertoire */pits/public* contient un fichier htaccess dont le contenu est placé ci-dessous.
8. L'application utilise l'upload de fichiers pour mettre à jour la table *elevés* après importation dans BdTS. Les fichiers uploadés sont placés dans */tmp*. Il est possible de modifier ce paramètre dans *config.ini*, mais il est conseillé de choisir un répertoire hors de la zone web.

### Contenu du fichier *.htaccess* du répertoire *pits/public*

```

1 RewriteEngine On
2 RewriteCond %{REQUEST_FILENAME} -s [OR]
3 RewriteCond %{REQUEST_FILENAME} -l [OR]
4 RewriteCond %{REQUEST_FILENAME} -d
5 RewriteRule ^.*$ - [NC,L]
6 RewriteRule ^.*$ index.php [NC,L]
7 #
8 Options -Indexes
9 #
10 #
11 php_flag magic_quotes_gpc off
12 php_flag register_globals off

```

## 2.4 - Fichiers de configuration

Il y a deux types de fichiers de configuration :

- les fichiers de configuration de l'application sont des fichiers ini ;
- les fichiers modèles des mails et des pdf sont des fichiers txt.

### 2.4.1 - Description des fichiers ini

#### 2.4.1.1 - Fichier config.ini

Le fichier contient trois sections. La section [app] est commune à l'ensemble. La section [dev] hérite de [app] et définit la configuration de développement. La section [prod] définit la configuration de production.

C'est dans cette dernière section que vous devez placer les paramètres connexion à la base de donnée.

```

1  ;; Configuration générale (adapter ce qui est en rouge sur fond jaune)
2
3  [app]
4  annee.scolaire           = 2010-2011
5  database.adapter         = pdo_mysql
6  database.params.dbname   = nom_de_la_base
7  mail.from.email          = nepasrepondre@cc-montesquieu.fr
8  mail.from.name           = Transports scolaires CCM
9  mail.replyto.email       = transport-scolaire@cc-montesquieu.fr
10 mail.replyto.name        = CCM - Transports scolaires
11 mail.body.text           = 1
12 mail.body.html           = 1
13
14 [dev : app]
15 database.params.host      = localhost
16 database.params.username  = mon_user_developpeur
17 database.params.password  = mon_mot_de_passe_developpeur
18 logfile                  = php://output
19 debug                    = 1
20 mail.replyto.email        = dafap@free.fr
21 mail.replyto.name         = PiTS Dev - Transports scolaires CCM
22 mail.transport            = smtp
23 uploadsavpath            = d:/tmp
24 sadmin                   = mon_email_sadmin
25
26 [prod : app]
27 database.params.host      = localhost
28 database.params.username  = nom_user_de_la_base
29 database.params.password  = mot_de_passe_de_user
30 logfile                  = /logs/log.log
31 debug                    = 0
32 mail.transport            = sendmail
33 uploadsavpath            = /tmp
34 sadmin                   = email_du_sadmin

```

#### Note de configuration :

Le choix entre [dev] et [prod] est réalisé dans *pits/public/index.php* par la constante APP\_MODE de la ligne 18

### 2.4.1.2 - Fichier menu.ini

Description des menus. Il y a 4 états :

- le simple visiteur, avant login
- le parent : user.categorie = 1
- l'administrateur : user.categorie = 2
- le super administrateur : user.categorie = 3

```

1  ;;
2  ;; Description des menus
3  ;;
4  [general]
5  url-retour = http://www.cc-montesquieu.fr/v2/index.php/content/view/1111173/897/
6
7  [visiteur : general]
8  ;;urlRetour = retour
9  index = accueil
10 creation = "création d'un compte"
11 help = aide
12
13 [parent : general]
14 logout = déconnexion
15 inscriptionleveve = "ajout d'un élève"
16 modifcompte = "modifier son compte"
17 help = aide
18
19 [admin : general]
20 logout = déconnexion
21 index = administrateur
22 listusers = "utilisateurs"
23 listelevess = "élèves"
24 statistiques = statistiques
25 extraction = extraction
26 validinscriptions = validation
27 help = aide
28
29 [sadmin : general]
30 logout = déconnexion
31 index = accueil
32 editconfig = configuration
33 changemdp = "change mot de passe"

```

#### Note de configuration :

Il faudra peut-être adapter l'adresse de url-retour qui revient à la bonne page du site principal.

### 2.4.1.3 - Fichier session.ini

La section [prod] hérite de la configuration de la section [dev] en raccourcissant le temps de vie de la session.

```
1  ;;
2  ;; Configuration de la session
3  ;;
4
5  [dev]
6  ; bug_compat_42
7  ; bug_compat_warn
8  ; cache_expire
9  ; cache_limiter
10 ; cookie_domain
11 ; cookie_lifetime
12 ; cookie_path
13 ; cookie_secure
14 ; entropy_file
15 ; entropy_length
16 ; hash_bits_per_character
17 ; hash_function
18 ; referer_check
19 ; save_handler
20 ; serialize_handler
21
22 use_cookies           = on
23 use_only_cookies     = on
24 use_trans_sid        = off
25 strict               = off
26 remember_me_seconds = 0
27 name                 = pits_session
28 gc_divisor           = 10
29 gc_maxlifetime       = 86400
30 gc_probability       = 1
31 save_path            = /tmp
32
33 [prod : dev]
34 remember_me_seconds = 0
35 gc_divisor          = 1000
36 gc_maxlifetime      = 600
37 gc_probability      = 1
```

**Note aux développeurs :**

save\_path est préfixé par le chemin complet de pits/application  
(voir ligne 107 de public/index.php)



## 2.4.2 - Description des fichiers txt

### 2.4.2.1 - Fichier mail\_eleve.txt

Corps de message du mail adressé au parent qui vient d'inscrire un enfant.

```

1  Bonjour et bienvenue à la Communauté de Communes de Montesquieu.
2  Vous venez d'inscrire un enfant au service des transports scolaires.
3  Vous trouverez en fichier joint le justificatif de cette inscription.
4  Vous devez en adresser une copie avec une photo d'identité de l'enfant et, si
5  vous avez choisi de payer par prélèvement automatique, un relevé d'identité
6  bancaire à l'adresse :
7  Communauté de communes de Montesquieu
8  Service des transports scolaires
9  1 allée Jean Rostand
10 33650 MARTILLAC
11 Tél. 05 57 96 79 62
12 Email: transport-scolaire@cc-montesquieu.fr

```

### 2.4.2.2 - Fichier mail\_user\_debloque.txt

Corps de message du mail adressé au parent lorsque l'administrateur valide son compte pour le débloquent.

```

1  $bloc$
2  Bienvenue à la Communauté de Communes de Montesquieu.
3  $finbloc$
4  $bloc$
5  Votre compte d'accès au service des transports scolaires vient d'être validé
6  par l'administrateur du service. Vous pouvez désormais vous connecter à partir
7  de la page d'accueil à l'adresse $url$.
8  -----
9  Votre login est : $Email$
10 Si vous ne connaissez pas votre mot de passe, vous pouvez en demander un
11 nouveau depuis la page d'accueil par le lien "Oubli du mot de passe".
12 -----
13 $finbloc$
14 $bloc$
15 Voici ce que contient votre fiche :
16 $finbloc$
17 $bloc$
18 Nom      : $Titre$ $Nom$ $Prenom$
19 Adresse  : $AdresseL1$
20           $AdresseL2$
21           $CodePostal$ $Commune$
22 Email    : $Email$
23 Téléphone : $liste(Telephone,TelephoneC)$
24 $finbloc$
25 $bloc$
26 Date de création du compte : $dateFR(dd/MM/YYYY HH:mm:ss,dateCreation)$
27 Date de modification       : $dateFR(dd/MM/YYYY HH:mm:ss,dateModif)$
28 Date de dernière connexion : $dateFR(dd/MM/YYYY HH:mm:ss,datePreviousLogin)$
29 $finbloc$
30 $bloc$
31 En cas d'anomalie, merci de bien vouloir prévenir le service.
32   Communauté de communes de Montesquieu

```

```

33         Service des transports scolaires
34             1 allée Jean Rostand
35             33650 MARTILLAC
36         Tél. 05 57 96 79 62
37         Email: transport-scolaire@cc-montesquieu.fr
38 $finbloc$

```

### 2.4.2.3 - Fichier mail\_user\_modif.txt

Corps de message du mail adressé au parent qui vient de modifier son compte.

```

1  $bloc$
2  Bienvenue à la Communauté de Communes de Montesquieu.
3  $finbloc$
4  $bloc$
5  Votre compte d'accès au service des transports scolaires vient d'être modifié.
6  -----
7  login          : $Email$
8  mot de passe   : $mdp$
9  -----
10 $finbloc$
11 $bloc$
12 Voici ce que contient votre fiche :
13 $finbloc$
14 $bloc$
15 Nom           : $Titre$ $Nom$ $Prenom$
16 Adresse       : $AdresseL1$
17                $AdresseL2$
18                $CodePostal$ $Commune$
19 Email         : $Email$
20 Téléphone     : $liste(Telephone,TelephoneC)$
21 $finbloc$
22 $bloc$
23 Date de création du compte : $dateFR(dd/MM/YYYY HH:mm:ss,dateCreation)$
24 Date de modification      : $dateFR(dd/MM/YYYY HH:mm:ss,dateModif)$
25 Date de dernière connexion : $dateFR(dd/MM/YYYY HH:mm:ss,datePreviousLogin)$
26 $finbloc$
27 $bloc$
28 En cas d'anomalie, merci de bien vouloir prévenir le service.
29     Communauté de communes de Montesquieu
30     Service des transports scolaires
31     1 allée Jean Rostand
32     33650 MARTILLAC
33     Tél. 05 57 96 79 62
34     Email: transport-scolaire@cc-montesquieu.fr
35 $finbloc$

```

### 2.4.2.4 - Fichier mail\_user\_premier.txt

Corps de message du mail adressé au parent qui vient de créer son compte.

```

1  $bloc$
2  Bonjour $Email$,
3  Bienvenue au service des transports scolaires de la Communauté de Communes de
4  Montesquieu.
5  $finbloc$

```

```

6      $bloc$
7      Pour éviter la création non autorisée d'un compte en votre nom, nous avons mis
8      en place un dispositif nécessitant votre validation.
9      Si vous n'avez pas demandé à ouvrir ce compte, nous vous recommandons de
10     suivre les instructions pour l'annuler.
11     $finbloc$
12     $bloc$
13     CONFIRMER LE COMPTE
14     Pour confirmer cette demande d'ouverture de compte, sélectionnez et copiez le
15     lien suivant. Ouvrez un navigateur et collez le lien dans la barre d'adresses.
16     Appuyez sur la touche Entrée ou Retour de votre clavier. Votre compte sera
17     validé et vous entrerez dans le service.
18         $url$/login/validation/e/$Email$/t/$temoin$
19     $finbloc$
20     $bloc$
21     ANNULER LE COMPTE
22     Si vous n'avez pas demandé l'ouverture de ce compte avec cette adresse de
23     messagerie et si vous souhaitez annuler cette demande, sélectionnez et copiez
24     le lien suivant. Ouvrez un navigateur et collez le lien dans la barre
25     d'adresses. Appuyez sur la touche Entrée ou Retour de votre clavier.
26         $url$/login/annulation/e/$Email$/t/$temoin$
27     $finbloc$
28     $bloc$
29     IMPORTANT
30     Pour vous aider à sécuriser vos informations personnelles, la Communauté de
31     Communes de Montesquieu recommande de ne jamais entrer vos informations
32     d'identification (adresse de messagerie et mot de passe) sur un site auquel
33     vous avez accédé en cliquant sur un lien intégré à un message électronique. Au
34     lieu de cliquer sur le lien, copiez l'adresse et collez-la dans la barre
35     d'adresses de votre navigateur Web. Nous vous envoyons parfois des messages
36     électroniques contenant des liens, mais ces derniers ne sont fournis que pour
37     des raisons pratiques.
38     $finbloc$
39     $bloc$
40     En cas d'anomalie, merci de bien vouloir prévenir le service.
41         Communauté de communes de Montesquieu
42         Service des transports scolaires
43         1 allée Jean Rostand
44         33650 MARTILLAC
45         Tél. 05 57 96 79 62
46         Email: transport-scolaire@cc-montesquieu.fr
47     $finbloc$

```

#### 2.4.2.5 - pdf\_eleve.txt

Modèle de création du justificatif d'inscription.

```

1      $bloc$
2      Nom : $Nom$
3      Prénom : $Prenom$
4      $rem: -----
5      $rem: Lorsqu'on utilise l'instruction dateFR(format, champ) il ne faut pas
6      mettre d'autres champs spéciaux dans la ligne
7      Date de naissance : $dateFR(dd/MM/YYYY, DateN)$
8      $rem: -----
9      Etablissement scolaire fréquenté : $NomEcole$ - $VilleEcole$
10     Régime : $Regime$
11     Classe : $Classe$
12     Responsable légal : $TitreR1$ $NomR1$ $PrenomR1$

```

```

13 Adresse : $AdressR1L1$
14           $AdressR1L2$
15           $CodePostalR1$ $NomVilleR1$
16 $rem: -----
17 $rem: L'instruction liste(...) doit être précédée de la définition du
18 séparateur
19 $rem: Lorsqu'on utilise l'instruction liste(...) il ne faut pas mettre
20 d'autres champs spéciaux dans le ligne
21 $separateur( - )$
22 Téléphone : $liste(TelephoneR1,TelephoneR1c)$
23 $rem: -----
24 Email : $Email$
25 Point de montée : $VilleStationR1$ - $NomStationR1$
26 Tarif : $TarifLibelle$ - $TarifTarif$ €
27 $finbloc$
28 $rem: -----
29 $rem: Une conditionnelle if (...) ne contient rien d'autre sur la ligne
30 $if (TarifPrelevement)$
31 $rem: -----
32 $bloc$
33 Relevé d'identité bancaire
34 $RibBanque$ $RibAgence$ $RibCompte$ $RibCle$
35 Banque : $RibDom$
36 Titulaire : $RibTit$
37 $finbloc$
38 $rem: -----
39 $rem: L'instruction endif ne contient rien d'autre sur la ligne
40 $endif$
41 $rem: -----
42 $if (SecondeAdresse)$
43 $bloc$
44 Garde alternée
45 Autre responsable légal : $TitreR2$ $NomR2$ $PrenomR2$
46 Adresse : $AdressR2L1$
47           $AdressR2L2$
48           $CodePostalR2$ $NomVilleR2$
49 $rem: -----
50 $rem: Ici, on utilise le même séparateur que celui défini plus haut pour la
51 première liste
52 Téléphone : $liste(TelephoneR2,TelephoneR2c)$
53 Point de montée : $VilleStationR2$ - $NomStationR2$
54 $finbloc$
55 $endif$
56 $rem: -----
57 $hr$
58 $bloc$
59 Adressez une copie de ce justificatif avec :
60   - une photo d'identité de l'élève
61 $if (TarifPrelevement)$
62   - un relevé d'identité bancaire
63 $endif$
64 à l'adresse :
65 $centre$
66 Communauté de Communes de Montesquieu
67 Service des transports scolaires
68 1 allée Jean Rostand
69 33650 MARTILLAC
70 $fincentre$
71 $finbloc$

```

## 2.4.3 - Langage pour fichier txt

### 2.4.3.1 - Éléments du langage

#### 2.4.3.1.1 - Les champs de tables et des vues de la base de donnée

Le nom des champs est encadré par le caractère § (paragraphe)

Les tables et vues traitées actuellement sont **eleveslist** et **user**.

#### 2.4.3.1.2 - Syntaxe

Les lignes commençant par **\$rem:** et finissant par **\$** sont des remarques et ne sont pas rendues. Elles ne créent pas de ligne vide dans le rendu.

Les lignes comprises entre une instruction **\$if(condition)\$** et une instruction **\$endif\$** ne sont interprétées que si le booléen *condition* est vrai.

Les lignes comprises entre une instruction **\$bloc\$** et une instruction **\$finbloc\$** sont interprétées comme un paragraphe (comme les balises <p> et </p> du html).

Les instructions conditionnelles **\$if(field)\$** et **\$endif** ne contiennent rien d'autre sur la ligne.

Les instructions **\$liste(field,[field2, ...])\$** ne contiennent pas d'autres instructions ou d'autres **\$champ\$** sur la ligne.

Pour l'instruction **\$dateFR(format,date)\$**, voici la syntaxe du format :

- d pour le jour sur 1 ou 2 chiffres,
- dd pour le jour en 2 chiffres,
- ddd le nom abrégé du jour,
- dddd le nom complet du jour
- M pour le mois sur 1 ou 2 chiffres,
- MM pour le mois sur 2 chiffres,
- MMM pour le nom abrégé du mois,
- MMMM pour le nom du mois complet
- YY pour l'année sur 2 chiffres,
- YYYY pour l'année en abrégé
- HH pour l'heure sur 24
- H pour l'heure sur 12
- mm pour les minutes
- ss pour les secondes

Le format lui même est libre. Il ne doit pas être mis entre guillemets ou apostrophes.

## 2.4.3.2 - Interpréteur et classes dérivées

### 2.4.3.2.1 - Pits\_Document\_Structure

L'interpréteur du langage est la classe *Pits\_Document\_Structure* du fichier :

*pits/library/Pits/Document/Structure.php*

Le constructeur de cette classe ouvre le document modèle (document .txt)

La classe possède les méthodes *feof()* et *fgets()* qui permettent de construire le document final ligne par ligne, en lisant chaque ligne du document modèle et en interprétant les éléments suivants :

- **\$rem: ... \$**
- **\$field\$** où *field* est le nom d'un champ d'une table
- **\$dateFR(format,field)\$**
- **\$separateur( - )\$**  
 Cette instruction précise le séparateur utilisé pour interpréter l'instruction \$liste(...)\$\$. Le séparateur sera tout ce qui se trouve entre parenthèses, espaces compris.  
 Cette instruction doit précéder l'instruction \$liste(...)\$\$.
- **\$liste(field1,[field2, ...])\$**
- **\$if(field)\$** où *field* est un champ qui sera interprété comme un booléen (0: faux ; autre chose : vrai)
- **\$endif\$**

### 2.4.3.2.2 - Les classes dérivées : PdfEleve et UserMailContent

Ces classes sont placées dans le répertoire *pits/application/models*. Elles sont appelées par des aides d'action du répertoire *pits/library/Pits/Controller/ActionHelpers*.

Elles chargent la table de donnée ou la vue MySQL à traiter et interprètent à leur tour les valeurs particulières qui sont définies dans ces classes. Elles interprètent également les instructions **\$bloc\$** et **\$finbloc\$**.

Le constructeur appelle le constructeur parent et charge les tables *\_tableFields* et *\_tableValues*, ainsi que les valeurs particulières comme le *mdp* par exemple.

La méthode *run()* lance la boucle d'interprétation et renvoie une valeur attendue par l'aide d'action.

#### 2.4.3.2.2.1 - Valeurs particulières de PdfEleves

- **\$regime\$** (décodage du champ)
- **\$hr\$** (ligne horizontale)
- **\$centre\$**
- **\$fincentre\$**

#### 2.4.3.2.2.2 - Valeurs particulières de UserMailContent

- **\$mdp\$** (mot de passe en clair ou valeur « inchangé »)
- **\$url\$** (adresse de connexion pour la validation du compte)

## 2.5 - Fichiers de logs

---

Le fichier log de l'application est enregistré sous le nom indiqué dans la ligne *logfile* du fichier *config.ini*.

**Note aux développeurs :**

logfile est préfixé par le chemin complet de pits/application  
(voir ligne 91 de *public/index.php*)

Il peut être récupéré par connexion ftp et supprimé. Un nouveau fichier sera alors ouvert.

**IMPORTANT !**

Ce fichier doit être surveillé afin qu'il ne devienne pas trop gros !

Le fichier de log contient les erreurs de l'application ainsi que la trace des login des utilisateurs.

## 2.6 - Fichiers de documentation

---

Les documents d'aide et les documents relatifs au service et à mettre à disposition des utilisateurs doivent être déposés dans le répertoire *pits/public/documents*. Seuls, les documents au format *.doc* ou *.pdf* peuvent être mis en ligne par l'administrateur de l'application. Votre webmestre pourra mettre en ligne d'autres types de documents en utilisant son client ftp.

## 3 - Organisation de l'application

### 3.1 - Les vues de l'application

#### 3.1.1 - Organisation, images et logo

L'organisation des vues de l'application est définie par :

*pits/application/views/layouts/layout.phtml*

Le bandeau du haut des pages est le fichier :

*pits/public/images/bandeau\_transports\_scolaires.jpg*

Le logo de l'organisateur est le fichier :

*pits/public/images/bandeau\_logo.png*

#### 3.1.2 - Statistiques de fréquentation des pages

Le code permettant de tenir à jour des statistiques doit être placé dans :

*pits/application/views/scripts/common/stats.phtml*

### 3.2 - Les controllers

#### Note aux développeurs :

Les controllers sont des classes dérivées de la classe `Pits_Controller_Action` qui définit 2 méthodes :

- méthode `getBaseUrl()` ;
- méthode `setMenu($user)` où `$user` est le nom de la section utilisateur définie dans `menu.ini`.

#### 3.2.1 - Accès anonyme : IndexController.php

##### 3.2.1.1 - Page d'accueil - indexAction

Donne la page d'accueil présentant le formulaire de connexion au service.

Cette page est entièrement décrite dans la vue *pits/application/views/scripts/index/index.phtml*

##### 3.2.1.2 - Création d'un compte - createAction

Gestion de la création d'un compte utilisateur. Cette page utilise le formulaire décrit dans *pits/library/Pits/Form/CreationParent.php* ainsi que deux vues :

- *pits/application/views/scripts/index/creation.phtml*
- *pits/application/views/scripts/index/creationmsg.phtml*



### 3.2.1.3 - Aide - helpAction

Gestion de la page de présentation des documents mis en ligne par l'organisateur. Cette page utilise l'aide d'action *pits/library/Pits/Controller/ActionHelpers/ListeDocuments.php* et le code factorisé de la vue *pits/application/views/scripts/common/listedocs.phtml*.

### 3.2.1.4 - Envoi d'un mot de passe - passwdAction

Permet de se faire envoyer un mot de passe si on l'a oublié ! Cette page utilise le formulaire décrit dans *pits/library/Pits/Form/NewPassword.php* ainsi que deux vues :

- *pits/application/views/scripts/index/passwd.phtml*
- *pits/application/views/scripts/index/creationmdp.phtml*

## 3.2.2 - Accès parent : ParentController.php

L'accès à ce contrôleur est réservé aux utilisateurs identifiés.

### 3.2.2.1 - Page d'accueil - indexAction

Cette page présente la liste des enfants inscrits par cet utilisateur.

Si les fiches des enfants n'ont pas été extraites, 3 boutons apparaissent : Modifier, Supprimer, Editer le justificatif.

Lorsqu'une fiche a été extraite (voir 3.2.3.15), le message Demande en cours de traitement est présent sur sa ligne et 2 boutons apparaissent : Signaler un changement, Editer le justificatif.

Lorsque l'inscription d'un élève dans BdTS est confirmée (voir 3.2.3.16), le message Inscription validée est présent sur la ligne de cet élève ainsi que 2 boutons : Signaler un changement, Editer le justificatif.

### 3.2.2.2 - Inscription d'un enfant - inscriptioneleveAction

Cette page présente le formulaire d'inscription et traite le résultat. Elle fait appel

- à l'aide d'action *pits/library/Pits/Controller/ActionHelpers/EditEleve.php*
- au formulaire *pits/library/Pits/Form/InscriptionEleve.php*
- au fichier *javascript pits/public/js/arrayPHP2JS.js*
- au fichier *javascript pits/public/js/formElv.js*

### 3.2.2.3 - Modification d'une inscription - editeleveAction

Cette page présente le formulaire d'inscription et traite le résultat. Elle fait appel exactement aux mêmes composants que la précédente :

- à l'aide d'action *pits/library/Pits/Controller/ActionHelpers/EditEleve.php*
- au formulaire *pits/library/Pits/Form/InscriptionEleve.php*
- au fichier *javascript pits/public/js/arrayPHP2JS.js*
- au fichier *javascript pits/public/js/formElv.js*

Seule l'initialisation est différente.

#### **3.2.2.4 - Supprime un enfant - suppreleveAction**

Cette action se fait après confirmation. Un contrôle d'identité du parent est effectué avant la suppression afin qu'un parent ne puisse pas supprimer l'enfant d'un autre.

#### **3.2.2.5 - Envoie un pdf de la fiche élève par mail - maileleveAction**

Cette action n'a pas de vue et transmet la fiche de l'élève sous format pdf par mail et par le navigateur. Un contrôle d'identité du parent est effectué avant la création de la fiche pour qu'un parent ne puisse pas voir la fiche de l'enfant d'un autre.

#### **3.2.2.6 - Modifier son compte - modifcompteAction**

Permet à l'utilisateur identifié de modifier son compte. Cette page utilise le formulaire décrit dans *pits/library/Pits/Form/CreationParent.php* .

#### **3.2.2.7 - Aide - helpAction**

Idem 3.2.1.3

#### **3.2.2.8 - Déconnexion - logoutAction**

Appelle le logout du controleur *LoginController*. Voir 3.2.5.5.

### **3.2.3 - Accès administrateur : AdminController.php**

L'accès à ce contrôleur est réservé aux utilisateurs identifiés dont la catégorie est supérieure à 1.

Un administrateur a accès au *ParentController*.

#### **3.2.3.1 - Page d'accueil - indexAction**

Cette page est décrite dans la vue :

*pits/application/views/scripts/admin/index.phtml*

#### **3.2.3.2 - Affiche la liste de utilisateurs - listusersAction**

Présente la liste des utilisateurs dans l'ordre alphabétique à partir du premier utilisateur correspondant à un critère de recherche.

Les critères portent sur le nom (début de nom) et sur l'email (partie d'un email, pas forcément le début). ils sont saisis à partir du formulaire *Pits\_Form\_Recherche* décrit dans le fichier :

*pits/library/Pits/Form/Recherche.php*

Un paginateur est utilisé. Son aspect est décrit dans le partial :

*pits/application/views/scripts/common/pagination\_control.phtml*

Le format de la liste est décrit dans la vue

*pits/application/views/scripts/admin/listusers.phtml*

Cette vue est assez complexe.

- Le bouton *Validation* apparaît lorsque le compte est bloqué (bloque = 1). Tous les autres boutons sont alors cachés.
- Le bouton *Changer* sert à changer le mot de passe
- Le bouton *Supprimer* n'est présent que si cet utilisateur n'a pas inscrit d'enfant
- Si l'utilisateur a inscrit des enfants, un bouton indiquant le nombre d'enfants permet d'obtenir la liste des enfants de cet utilisateur (action *voirenfantsAction*)

### **3.2.3.3 - Débloque un compte - debloqueAction**

Valide un compte d'utilisateur et envoi un email de confirmation. Cette page utilise l'aide d'actio *UserMail* décrite dans

*pits/library/Pits/Controller/ActionHelpers/UserMail.php*

et le modèle *UserMailContent* décrit dans

*pits/application/models/UserMailContent.php*

Le corps du message est obtenu à partir du fichier modèle

*pits/application/config/mail\_user\_debloque.txt*

### **3.2.3.4 - Change un mot de passe - changempAction**

Change le mot de passe d'un utilisateur et lui envoie par email. Cette page utilise le formulaire

*pits/library/Pits/Form/ChangeMdp.php*

Cette action utilise l'aide d'action *UserMail* décrite dans

*pits/library/Pits/Controller/ActionHelpers/UserMail.php*

et le modèle *UserMailContent* décrit dans

*pits/application/models/UserMailContent.php*

Le corps du message est obtenu à partir du fichier modèle

*pits/application/config/mail\_user\_modif.txt*

### **3.2.3.5 - Supprime un utilisateur - suppruserAction**

Supprime un utilisateur qui n'a pas inscrit d'enfant, après confirmation.

Utilise le formulaire *Pits\_Form\_Confirmation* défini dans le fichier :

*pits/library/Pits/Form/Confirmation.php*

La vue est unique et prend des allures différentes selon le cas. Voir le fichier :

*pits/application/views/scripts/admin/suppruser.phtml*

### **3.2.3.6 - Affiche le liste des élèves - listelevsAction**

Présente la liste des 500 premiers élèves dans l'ordre alphabétique à partir du premier élève correspondant à un critère de recherche.

Les critères portent sur le nom (début de nom) et sur l'email (partie d'un email, pas forcément le début). ils sont saisis à partir du formulaire *Pits\_Form\_Recherche* décrit dans le fichier :

*pits/library/Pits/Form/Recherche.php*

Un paginateur est utilisé. Son aspect est décrit dans le partial :

*pits/application/views/scripts/common/pagination\_control.phtml*

Le format de la liste est décrit dans la vue

*pits/application/views/scripts/admin/listeleves.phtml*

### **3.2.3.7 - Voir la liste des enfants d'un utilisateur - voirenfansAction**

Présente la liste des enfants d'un utilisateur.

Le format de la liste est décrit dans la vue

*pits/application/views/scripts/admin/voirenfans.phtml*

### **3.2.3.8 - Modification d'une fiche d'élève - editeleveAction**

Cette page présente le formulaire d'inscription et traite le résultat. Elle fait appel

- à l'aide d'action *pits/library/Pits/Controller/ActionHelpers/EditEleve.php*
- au formulaire *pits/library/Pits/Form/InscriptionEleve.php*
- au fichier *javascript pits/public/js/arrayPHP2JS.js*
- au fichier *javascript pits/public/js/formElv.js*

Elle a un traitement commun aux inscriptions et modifications d'élèves par les parents. Elle ne peut être appelée qu'à partir de la liste des élèves depuis *admin/listelevesAction* ou *admin/voirelevesAction*.

### **3.2.3.9 - Supprime un élève - suppreleveAction**

Supprime la fiche d'un élève après confirmation.

Utilise le formulaire *Pits\_Form\_Confirmation* défini dans le fichier :

*pits/library/Pits/Form/Confirmation.php*

### **3.2.3.10 - Crée un pdf de la fiche élève et l'envoie à l'écran - pdfleveAction**

Cette action n'a pas de vue et transmet la fiche de l'élève sous format pdf par le navigateur. Un contrôle d'identité du parent est effectué avant la création de la fiche pour qu'un parent ne puisse pas voir la fiche de l'enfant d'un autre.

### **3.2.3.11 - Page d'accès aux statistiques - statistiquesAction**

Présente une page d'accès aux statistiques prévues. La liste des états de statistiques prévus est écrite dans la vue :

*pits/application/views/scripts/admin/statistiques.phtml*

### **3.2.3.12 - Statistiques par point d'arrêt - statparstationAction**

Affichage d'une requête. Aucune difficulté.

### **3.2.3.13 - Statistiques par établissement - statparetabAction**

Affichage d'une requête. Aucune difficulté.

### 3.2.3.14 - Statistiques par commune - statparvilleAction

Affichage d'une requête. Aucune difficulté.

### 3.2.3.15 - Extraction csv pour BdTS - extractionAction

Cette action transmet un fichier au format .csv d'extraction des fiches élèves créés ou modifiés à partir d'une date donnée. Les données du fichier sont séparées par des ; et encadrées par des ".

La page présente le formulaire Pits\_Form\_ExtractionDepuis décrit dans le fichier :

*pits/library/Pits/Form/ExtractionDepuis.php*

Le formulaire est pré-rempli par la date et l'heure de la dernière extraction. Il est possible de donner une date antérieure si on veut relancer une extraction.

L'action suit l'algorithme suivant :

- fixe la date au début du traitement
- marque les fiches concernées par cette date dans le champ dateExtraction
- compte le nombre d'élèves à extraire
- crée un flux temporaire pour constituer le fichier .csv
- transmet ce flux en téléchargement
- marque les fiches concernées comme extraite (ficheExtraite = 1) et non modifiées (ficheModifiée = 0)

### 3.2.3.16 - Confirmation des inscriptions dans BdTS - validinscriptionsAction

Cette action upload le fichier extrait de **Pits2BdTS** pour marquer les fiches élèves avec la mention « Inscription validée » dans la page des parents.

Le fichier à remonter provient du logiciel **Pits2BdTS** et le dispositif utilise le formulaire *Pits\_Form\_ValidInscriptions* décrit dans le fichier :

*pits/library/Pits/Form/ValidInscription.php*

Le fichier est contrôlé avant d'être traité. Il contient des requêtes SQL du type :

```
1 UPDATE eleves SET CodeStationR1=270, CodeStationR2=0, ficheValidee=1,
2 dateValidation="2010-07-27 16:44:38" WHERE eleveId=10;
```

Ce modèle est déposé dans le fichier validateur :

*pits/library/Pits/Validate/ControleFileValidation.php*

dans la constante DEFAULT\_MODEL.

#### Note aux développeurs :

Ce modèle doit être adapté si l'extraction de Pits2BdTS vient à être modifiée. Il suffit alors de copier coller une nouvelle ligne.

*Le contrôle porte sur la vérification du nombre de caractères autres que les chiffres. Les caractères non imprimables sont vérifiés.*

*La mise à jour ne contient pas de caractère accentués afin qu'il n'y ait pas de problème de codage utf\_8 (access produisant un fichier encodé iso\_8859\_1).*

### 3.2.3.17 - Aide - helpAction

Idem 3.2.1.3 mais avec le bouton 'Nouveau document' et les boutons de suppression. Les boutons sont définis

- dans la vue *pits/application/views/scripts/admin/help.phtml* ;
- dans le partial *pits/application/views/scripts/common/listedocs.phtml*

Dans le partial *listedocs.phtml*, les boutons 'Supprimer' sont affichés lorsque la variable *\$btnSuppr* est définie. Pour cela, il suffit de la placer dans le tableau des paramètres d'appel du partial (voir *help.phtml*).

#### **3.2.3.18 - Met en ligne un document d'aide- addhelpAction**

Utilise le formulaire *Pits\_Form\_GestionHelp* défini dans *pits/library/Pits/Form/GestionHelp.php*. Ce formulaire contrôle le type du fichier (.doc et .pdf uniquement) et l'écrit dans *pits/public/documents*.

#### **3.2.3.19 - Supprime un document d'aide- supprhelpAction**

Supprime un fichier dans *pits/public/documents*.

#### **3.2.3.20 - Déconnexion - logoutAction**

Appelle le logout du contrôleur *LoginController*. Voir 3.2.5.5.

### **3.2.4 - Accès super administrateur : SAdminController.php**

L'accès à ce contrôleur est réservé aux utilisateurs identifiés dont la catégorie est égale à 3.

#### **3.2.4.1 - Page d'accueil - indexAction**

Propose des liens sur les espaces admin et parent et sur les autres actions du contrôleur.

#### **3.2.4.2 - Récupérer et vider le fichier log -razlogAction**

Télécharge le fichier log puis le vide après confirmation. Gestion de la configuration 'dev' où il n'y a pas de fichier log (php://output).

#### **3.2.4.3 - Gestion des administrateurs - admingestAction**

Affiche la liste des administrateurs et propose d'en ajouter, de les modifier, de les supprimer, de changer leur mot de passe.

#### **3.2.4.4 - Ajout d'un administrateur - adminaddAction**

Ajout d'un administrateur. Utilise le formulaire *Pits\_Form\_CreationParent* décrit plus haut.

#### **3.2.4.5 - Modification d'un administrateur - admineditAction**

Modification d'un administrateur. Utilise le formulaire *Pits\_Form\_CreationParent* décrit plus haut.

#### **3.2.4.6 - Suppression d'un administrateur - adminsupprAction**

Supprime, après confirmation, un administrateur qui n'a pas inscrit d'enfant.

Utilise le formulaire *Pits\_Form\_Confirmation* défini dans le fichier :

*pits/library/Pits/Form/Confirmation.php*

La vue est unique et prend des allures différentes selon le cas. Voir le fichier :

*pits/application/views/scripts/sadmin/adminsuppr.phtml*

#### **3.2.4.7 - Mot de passe d'un administrateur - adminmdpAction**

Change le mot de passe d'un administrateur et lui envoie par email. Cette page utilise le formulaire

*pits/library/Pits/Form/ChangeMdp.php*

Cette action utilise l'aide d'action *UserMail* décrite dans

*pits/library/Pits/Controller/ActionHelpers/UserMail.php*

et le modèle *UserMailContent* décrit dans

*pits/application/models/UserMailContent.php*

Le corps du message est obtenu à partir du fichier modèle

*pits/application/config/mail\_user\_modif.txt*

#### **3.2.4.8 - Vider la table des élèves - razelevesAction**

Vide la table élèves après confirmation.

#### **3.2.4.9 - Vider la table des utilisateurs - razuserAction**

Vide la table des utilisateurs après confirmation. Ne touche pas aux administrateurs et super administrateurs.

#### **3.2.4.10 - Upload et exécute un fichier sql - sqlAction**

Uploade un fichier de requêtes permettant de charger les fichiers annexes, puis l'exécute.

#### **3.2.4.11 - Décrit les tables - decritablesAction**

Affiche les infos sur les champs de chaque table de la base de donnée de l'application.

#### **3.2.4.12 - Liste des fichiers de configuration - editconfigAction**

Liste le répertoire *pits/application/config* et propose d'éditer les fichiers.

#### **3.2.4.13 - Edite un fichier du répertoire de configuration- editfichierAction**

Seuls, les fichiers *.ini* et *.txt* sont traités.

Pour les fichiers *.ini* seules les lignes de configuration sont modifiables. Pour les fichiers *.txt* l'édition se fait dans un *textarea*.

#### **IMPORTANT !**

L'année scolaire se trouve dans le fichier *config.ini*.

**Attention à la syntaxe de ces fichiers !**

#### 3.2.4.14 - Change le mot de passe de sadmin - changemdpAction

Change le mot de passe du super administrateur en demandant l'ancien mot de passe puis le nouveau mot de passe confirmé par une seconde frappe. Fait appel au formulaire *Pits\_Form\_ConfirmMdp* décrit dans le fichier :

*pits/library/Pits/Form/ConfirMdp.php*

Ce formulaire a une méthode `isValid()` intéressante.

#### 3.2.4.15 - Déconnexion - logoutAction

Appelle le logout du controleur *LoginController*. Voir 3.2.5.5.

### 3.2.5 - Gestion du login : LoginController.php

---

#### 3.2.5.1 - indexAction

Cette action renvoie vers la page d'accueil convenable pour l'utilisateur selon son état : anonyme, parent, administrateur ou super administrateur. Pas de vue.

#### 3.2.5.2 - Annulation de la création d'un compte - annulationAction

Cette action supprime l'utilisateur si le compte n'a jamais été validé. Sinon, refus.

Elle utilise deux vues :

*pits/application/views/scripts/login/annulation.phtml* en cas de succès

*pits/application/views/scripts/login/annulationrefus.phtml* si le compte est déjà validé.

#### 3.2.5.3 - Validation de la création d'un compte - validationAction

Cette action valide la création d'un compte en mettant le champ bloque à 0. Elle ne fonctionne que pour les comptes qui ne sont pas validés. Elle loge l'utilisateur et le place dans son espace parent ou admin (selon le cas).

Elle utilise la vue :

*pits/applications/views/scripts/login/validationrefus.phtml*

en cas de refus, si le compte est déjà validé.

#### 3.2.5.4 - Login - loginAction

Valide les données du formulaire de login puis,

- en cas d'échec, elle renvoie un message FlashMessenger à la page d'accueil pour indiquer l'erreur ;
- en cas de succès, elle charge en session les données de l'utilisateur puis l'oriente vers son espace, selon sa catégorie.

#### 3.2.5.5 - Logout - logoutAction

Détruit la session et redirige vers la page d'accueil de l'application.



## 3.3 - Gestion des erreurs

---

### 3.3.1 - Erreurs d'Url

---

Les erreurs d'Url sont gérées. Elles renvoient sur le contrôleur si l'erreur se trouve au niveau de l'action et sur l'entrée du site si l'erreur se trouve au niveau du contrôleur.

La gestion se fait par la méthode *preDispatch* de la classe *Pits\_Controller\_NavigationPlugin*. Ce plugin enregistre aussi en session l'Uri de la page précédente (méthode *dispatchLoopShutdown*) pour l'aide d'action *RedirectorToOrigin*.

### 3.3.2 - Interception des erreurs du programme

---

Les erreurs du programme sont interceptées par les classe de la librairie :

*pits/library/package.exceptions.errors.php*

Toutes les erreurs sont transformées en exception. Les exceptions prévues par le programmeur sont gérées. Les autres sont interceptées par le *bootstrapError* contenu dans *pits/public/index.php*.

Les erreurs sont tracées dans le fichier log de l'application. La verbosité dépend de la valeur du paramètre *debug* du fichier de configuration *pits/application/config/config.ini*.

### 3.3.3 - Gestion des exceptions - ErrorController

---

Les exceptions traitées par les librairies *library/Zend* et *library/Pits* sont gérées par le contrôleur *ErrorController*.

## 4 - Comptes d'administrateur et de super administrateur

### 4.1 - Comptes d'administrateurs

Date de création	Personne désignée	Email	Supprimé le
14/06/2010	Favary Mimi	transport-scolaire@cc-montesquieu.fr	
14/06/2010	Laborde Renaud	informatique@cc-montesquieu.fr	04/08/2010

### 4.2 - Comptes de super administrateurs

Date de création	Personne désignée	Email	Supprimé le
14/06/2010	Pomirol Alain	dafap@free.fr	
04/08/2010	Laborde Renaud	informatique@cc-montesquieu.fr	

**Achevé le 4 août 2010**

**Merci de signaler toute erreur à :**  
Alain POMIROL (dafap@free.fr)  
DAFAP Informatique  
42 av. Jeanne de Devant  
33210 LANGON