

ECOLE POLYTECHNIQUE DE L'UNIVERSITÉ FRANÇOIS RABELAIS DE TOURS

Département Informatique

64 avenue Jean Portalis

37200 Tours, France

Tél. +33 (0)2 47 36 14 14

polytech.univ-tours.fr

Projet Recherche & Développement 2017-2018

Outil de gestion de parcours patient dans un hôpital de jour

Tuteur académique
Yannick KERGOSIEN

Étudiant
Jing YANG (DI5)

2 avril 2018



Liste des intervenants

Nom	Email	Qualité
Jing YANG	jing.yang@etu.univ-tours.fr	Étudiant DI5
Yannick KERGOSIEN	yannick.kergosien@univ-tours.fr	Tuteur académique, Département Informatique



Avertissement

Ce document a été rédigé par Jing Yang susnommé l'auteur.

L'Ecole Polytechnique de l'Université François Rabelais de Tours est représentée par Yannick Kergosien susnommé le tuteur académique.

Par l'utilisation de ce modèle de document, l'ensemble des intervenants du projet acceptent les conditions définies ci-après.

L'auteur reconnaît assumer l'entière responsabilité du contenu du document ainsi que toutes suites judiciaires qui pourraient en découler du fait du non respect des lois ou des droits d'auteur.

L'auteur atteste que les propos du document sont sincères et assument l'entière responsabilité de la véracité des propos.

L'auteur atteste ne pas s'approprier le travail d'autrui et que le document ne contient aucun plagiat.

L'auteur atteste que le document ne contient aucun propos diffamatoire ou condamnable devant la loi.

L'auteur reconnaît qu'il ne peut diffuser ce document en partie ou en intégralité sous quelque forme que ce soit sans l'accord préalable du tuteur académique et de l'entreprise.

L'auteur autorise l'école polytechnique de l'université François Rabelais de Tours à diffuser tout ou partie de ce document, sous quelque forme que ce soit, y compris après transformation en citant la source. Cette diffusion devra se faire gracieusement et être accompagnée du présent avertissement.



Pour citer ce document

Jing Yang, *Outil de gestion de parcours patient dans un hôpital de jour*, Projet Recherche & Développement, Ecole Polytechnique de l'Université François Rabelais de Tours, Tours, France, 2017-2018.

```
@mastersthesis{
  author={Yang, Jing},
  title={Outil de gestion de parcours patient dans un hôpital de jour},
  type={Projet Recherche \& Développement},
  school={Ecole Polytechnique de l'Université François Rabelais de Tours},
  address={Tours, France},
  year={2017-2018}
}
```

Table des matières

Liste des intervenants	a
Avertissement	b
Pour citer ce document	c
Table des matières	i
Table des figures	v
I Recherche	1
1 Introduction	2
1 Introduction de projet	2
2 Contexte de la réalisation	2
2.1 Contexte	2
2.2 Description de problème	3
2.3 Identification du besoin.....	3
2.4 Définition des objectifs.....	3
2.5 Bases méthodologiques.....	4
2 Description générale	5
1 Environnement du projet	5
2 Caractéristique des utilisateurs.....	5
3 Contraintes de développement et d'exploitation.....	6
4 Description des interfaces externes du système	7
4.1 Interface matériel/ logiciel	7

4.2	Interfaces homme/ machine.....	7
4.3	Interface logiciel/ logiciel.....	7
5	Description des fonctionnalités	7
5.1	Revoir la suppression de la base de données	7
5.2	Créer une fonctionnalité PURGE	8
5.3	Améliorer la gestion des patients.....	8
5.4	Améliorer l'affectation d'un parcours.....	9
5.5	Faire valider l'activité	9
5.6	Refaire la partie de consultation sur les dossiers de parcours.....	9
5.7	Faire le plan de parcours correspondre au objectif de nombre de patients ..	9
6	Structure générale de système.....	9
3	Etat de l'art	11
1	Logiciels existants.....	11
2	Veille technologique	11
4	Analyse et conception	13
1	Conception des fonctionnalités.....	13
2	Analyse sur les fonctionnalités	13
2.1	Revoir la suppression de la base de données	13
2.2	Créer une fonctionnalité PURGE	13
2.3	Améliorer la gestion des patients.....	14
2.4	Améliorer l'affectation d'un parcours.....	14
2.5	Faire valider l'activité	14
2.6	Refaire la partie de consultation sur les dossiers de parcours.....	15
2.7	Faire le plan de parcours correspondre au objectif de nombre de patients ..	15
5	Bilan et conclusion	16
1	Tâches réalisées.....	16
2	Plan de développement	16
II	Développement	18
6	Méthodologie de suivi et gestion de projet	19
1	Suivi par l'encadrant	19
2	Versionning et gestion de projet.....	19
7	Mise en oeuvre	20
1	Fonctionnement général.....	20
2	Conception de la suppression	20

2.1	Principe	20
2.2	limite	20
3	Conception de gérer RDV.....	21
3.1	Principe	21
4	Conception de consulter dossiers parcours	21
4.1	Principe	21
8	Validation et tests	23
1	Tests unitaires	23
2	Tests fonctionnels	23
9	Avancement du projet	24
1	Fonctionnalités réalisées	24
2	Fonctionnalités reste à faire	24
10	Re-productivité	25
1	Pré-requis et installation	25
2	Déploiement	25
3	Documentation	25
11	Conclusion	27
Annexes		28
A	Maquettes détaillées	29
B	Maquettes détaillées	30
C	Les fonctionnalités déjà implémentées	31
1	Gérer les personnels	31
2	Gérer les ressources matérielles	31
3	Gérer les activités	31
4	Gérer les parcours.....	32
5	Gérer le plan des parcours	32
6	Visualiser le planning (patient).....	32
7	Affecter patient à un parcours	32
8	Visualiser et modifier le planning de toutes les ressources	32
D	Modèle conceptuel des données	33
E	Explications des tables de notre modèles	34

F	Gestion de projet	36
1	méthode de suivi de projet	36
2	Outil de versionning	36
3	Diagramme de Gantt	36
G	Guide développeur	38
1	Structure des sources de l'application	38
2	Contrôleurs	39
3	Modèles	40
4	Vues	40
H	Rapport de tests	41
I	Librairies utilisées	44
	Bibliographie	45



Table des figures

2 Description générale	
1 Diagramme de déploiement.....	6
2 Diagramme des cas d'utilisation	8
5 Bilan et conclusion	
1 Liste des tâches de notre projet.....	17
6 Méthodologie de suivi et gestion de projet	
1 Versionning du projet.....	19
7 Mise en oeuvre	
1 La gestion des RDVs.....	21
2 La consultation des dossiers.....	22
10 Re-productivité	
1 Documentation d'une classe	26
2 Documentation d'une méthode.....	26
A Maquettes détaillées	
1 Gestion de patient	29
B Maquettes détaillées	
1 Valider activité et consulter les dossiers parcours.....	30

D Modèle conceptuel des données

1	Diagramme de classes	33
---	----------------------------	----

F Gestion de projet

1	Diagramme de Gantt	37
---	--------------------------	----

G Guide développeur

1	La structure des sources d'application.....	38
2	Le répertoire app	39
3	L'exemple de contrôleur	39
4	L'exemple de modèle	40
5	Vue générale	40
6	Code changement de vue	40

H Rapport de tests

1	Tests sur les fonctionnalités	41
2	Tests sur les fonctionnalités	42
3	Tests sur les fonctionnalités	43

Première partie

Recherche

1

Introduction

1 Introduction de projet

Ce projet s'inscrit dans le cadre de mon projet de recherche et développement (PRD), au sein de l'école Polytech Tours, lors de la 5ème année d'étude au département informatique.

Le projet de gestion de parcours patients fait partie d'un ensemble de projets au sein de Polytech Tours. C'est un projet en deux aspects : la gestion et l'optimisation. Les deux projets sont sous l'égide de Yannick Kergosien, maître de conférences au département informatique de l'Ecole Polytechnique de Tours, et sous la tutelle de Lucie Roussel, ingénieur en ordonnancement à l'Assistance Publique - Hôpitaux de Paris (AP-HP) et maître d'ouvrage (MOA).

Le projet de gestion consiste au développement d'un système d'information comportant une interface, une base de données et une application mettant en œuvre les deux éléments précédents. Celle-ci a été développée par six étudiants au cours de leur projet SI et deux étudiants de leurs projets PRD. Cette application permet d'ajouter des patients, des ressources médicales, du personnel médical, de gérer les parcours et les activités médicales, et de planifier des parcours pour des patients.

Mon objectif dans ce projet est de reprendre le système, corriger les bugs, développer de nouvelles fonctionnalités, et de revoir la base de donnée.

Ce document est donc le cahier de spécification système du projet " La gestion de parcours de patient ". Il définit les besoins, l'environnement du projet et les objectifs à réaliser. Ce rapport présente aussi les différentes tâches à effectuer et le planning prévisionnel.

2 Contexte de la réalisation

2.1 Contexte

L'Assistance Publique – Hôpitaux de Paris (AP-HP) envisage un regroupement de différents services à l'hôpital Antoine Béclère (Clamart, 92) dans le cadre de son plan stratégique 2015-2019. Le regroupement de ces différents services permettra une optimisation des ressources et de l'espace sur le site.

Ce regroupement nécessite le développement d'un système d'information permettant la gestion des parcours patients (succession d'activités médicales) mais également le développement d'algorithmes permettant la planification des différents parcours.

2.2 Description de problème

Dans un hôpital du jour, les ressources matérielles et humaines sont limitées, il y a un besoin de réduire le gaspillage budgétaire pour maximiser la satisfaction des patients. Afin d'améliorer l'organisation des prises de rendez-vous, des outils numériques intégrant des techniques de la recherche opérationnelle peuvent être mis en place. Ces derniers permettent de gérer de meilleure façon l'utilisation des ressources médicales.

Pour désengorger les services hospitaliers écrasés sous la charge des patients venant pour des besoins différents avec des priorités différents, les centres de soins mettent en place de plus en plus des services dédiés à l'accueil de patients pour une matinée, un après-midi ou même durant une journée complète.

Les parcours cliniques sont des plans de soins pluridisciplinaires structurés exposant en détail les étapes essentielles que doivent suivre les patients présentant un problème clinique spécifique. Ceux-ci se décomposent en plusieurs activités prédéfinies, certaines devant être effectuées avant d'autres (contraintes de précedence), nécessitant des ressources multiples et diverses au sein d'un même service ou de plusieurs (contraintes de ressources), dans les délais et créneaux horaires journaliers (contraintes de délais, contraintes temporelles).

Un patient doit prendre un RDV par téléphone avant d'aller à l'hôpital et décrire son état donc le personnel de l'hôpital peut déterminer quel parcours lui convient. Selon les informations concernant la date et l'heure disponible que le patient fournit au téléphone, le personnel sera permis à planifier les activités dans le parcours pour le patient [1].

Dans la réalité, il existe peu d'outil qui répond aux besoins de ce dernier. Il n'y a pas encore un outil pour planifier des activités médicales automatiquement et en même temps permettre les patients à visualiser et suivre leurs plannings dans l'hôpital. Mais il y a des outils qui répondent une partie des problèmes et un outil créé pendant les anciens projets de recherche et de développement de notre école qui a réalisé certaines fonctionnalités (détaillé en Annexe). Donc notre projet est pour compléter et améliorer cet outil créé par les anciens étudiants.

2.3 Identification du besoin

Le but de ce projet est de compléter un outil de gestion de parcours patients en temps réel qui a été créé par les étudiants de leur PRD, d'ajouter de nouvelles fonctionnalités pour qu'il fonctionne bien et réduire les problèmes de fonctionnement. L'outil maintenant nous permet d'ajouter des données sur les patients, les ressources, de modifier les informations de ces derniers et également de visualiser les différents plannings de ces derniers.

2.4 Définition des objectifs

Mon projet est un projet de développement et on fera un rapide Etat de l'art des logiciels existants mais on sait qu'il y en a peu et en plus ils sont payants. Par contre on a un logiciel qui a été implémenté et qui va falloir le reprendre. Un des objectifs du projet sera donc après étudier les logiciels existants (voir Logiciels existants dans l'Annexe) et d'étudier en profondeur le fonctionnement du système fait par les étudiants de PRD pour produire des éléments qui sont manquants.

Une version du système d'information a déjà été développée par six étudiants de 5ème année au cours de l'année 2015-2016 et un étudiant de son projet de recherche et développement de l'année 2016-2017 qui répond une partie de problème et je vais travailler sur une autre partie. Les algorithmes ont également été développés par un autre étudiant dans le cadre de son projet de recherche et développement.

Mon travail consiste à reprendre ce système d'information, à corriger les bugs existant dans le système, à développer de nouvelles fonctionnalités et restructurer une partie de la base de données pour qu'elle puisse bien fonctionner et vérifier que les fonctionnalités montrées dans l'application peuvent fonctionner correctement.

2.5 Bases méthodologiques

Dans le cas du développement de ce système d'information, j'utilise une modélisation avec UML pour déterminer les cas d'utilisation et le diagramme de classe. Pour gérer l'avancement du projet, l'outil Trello sera utilisé depuis sa fonctionnalité Kanban qui nous permet de gérer les tâches.

Mon développement suivra un cycle de vie cascade et les sources du logiciels seront versionnées grâce à Git. L'architecture MVC (modèle, vue, contrôleur) intégré dans le Framework CodeIgniter sera utilisé.

2

Description générale

1 Environnement du projet

Ce projet est la reprise de trois projets existants, un projet de la gestion commencé par un groupe de 6 étudiants lors de leur 5ème année à l'école polytechnique de Tours de l'année 2015-2016 dont Jean Coquelet fait chef de projet, un projet de l'optimisation réalisé aussi par Jean Coquelet dans le cadre de son PRD de l'année 2015-2016 et un projet de développement fait par Pochet Guillaume de son PRD de l'année 2016-2017.

L'application existante est exécutée sur un serveur qui communiquera avec une base de données pour pouvoir récupérer les différentes informations de l'application. Un ordinateur ou smartphone (pour les patients) avec un navigateur sera demandé pour l'implémentation de cette application. Voici le diagramme de déploiement **Figure 1**.

Cette application existante a réalisé des fonctionnalités pour la gestion des ressources, des patients et des parcours, elle nous permet de planifier des parcours aux patients et de visualiser le planning d'un patient ou d'un personnel. Les fonctionnalités détaillées seront décrites dans la partie fonctionnalités déjà implémentées (voir Annexe).

Les pages webs avec des interfaces existent déjà. Ces interfaces utilisent tous les modules définis dans le diagramme de classe **Figure 1** (Annexe D). Dans mon PRD, je vais rajouter des éléments basés sur les interfaces existantes et modifier ce que n'a pas été bien développé.

Je vais revoir la suppression des éléments qui n'est pas bien résolu pendant l'ancien projet en restructurant une partie de la base de données. Pour la gestion de patient, je vais rajouter de nouvelles fonctionnalités et modifier les actions sur un patient cherché sur l'interface « gérer patient ». A part de ça, l'interface suivante qui est « dossier parcours » sera modifiée et les onglets et champs seront ajoutés ici. Selon l'interface « nouveau parcours », je vais retravailler sur la partie disponibilité qui ne changera pas l'interface.

2 Caractéristique des utilisateurs

Dans notre projet, on peut différencier 3 types d'utilisateurs de l'application web :

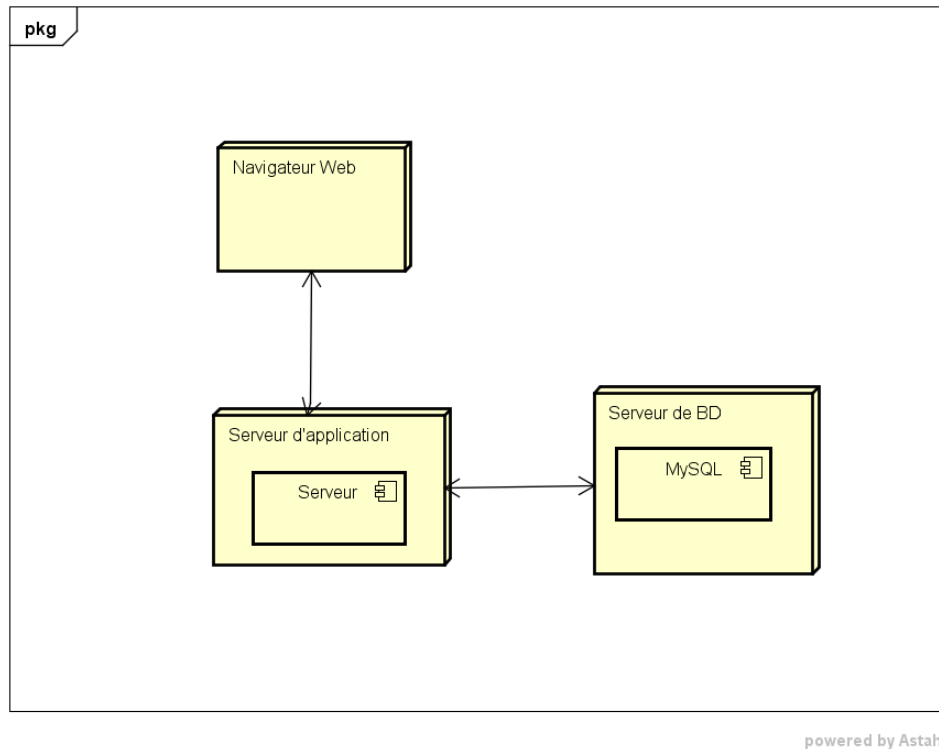


Figure 1 – Diagramme de déploiement

- Les Patients : Accès limité à l'application, ils pourront uniquement visualiser leur planning et suivre l'avancement dans leur parcours (sans indications horaires excepté sur la prochaine activité).
- Le personnel de soins : Ils auront accès à l'ajout / la recherche de patients, l'accès à leur planning et à la visualisation d'un planning d'un patient.
- Administrateurs et infirmières de coordination : C'est l'utilisateur qui a tous les droits sur l'application. Il peut visualiser/modifier les plannings de toutes les ressources, ajouter/modifier une activité, gérer les ressources (humaines ou matérielles), ajouter/modifier un parcours patient.

Les différents utilisateurs de l'application devront s'authentifier pour pouvoir accéder aux différentes fonctionnalités.

3 Contraintes de développement et d'exploitation

Pour la mise en place de l'application web, je vais reprendre les différents langages utilisés pour le développement du système d'information (HTML, CSS, JavaScript, PHP...). Je vais également suivre la même architecture de l'application (MVC) à l'aide du Framework PHP (CodeIgniter) et continuer à utiliser Bootstrap pour le CSS.

La réalisation des différents plannings sera réalisée à l'aide de la librairie FullCalendar/timeline.

Pour l'instant, l'exploitation n'est pas envisagée, donc il n'y a pas de contrainte d'exploitation.

4 Description des interfaces externes du système

4.1 Interface matériel/ logiciel

Dans ce projet, je dois faire des développements sur une interface web. Donc pour utiliser cette interface, on a besoin au moins un ordinateur ou un smartphone (pour les patients) avec un navigateur intégré (avec JavaScript activé). Pour la visualisation des différents plannings, l'activation de JavaScript est demandée. Cette interface communique avec une base de données pour récupérer les différentes informations de l'application (liste des ressources, parcours patients, planning, etc.).

4.2 Interfaces homme/ machine

Les utilisateurs de l'application auront tous un accès à l'application. L'authentification à l'aide d'un nom d'utilisateur et d'un mot de passe définira le type d'utilisateur.

Par conséquent, le type d'utilisateur définira les différents onglets ou pas à afficher.

Il existe 3 types d'utilisateurs (patient, ressource et administrateur ou infirmière de coordination) décrit dans la partie Caractéristique des utilisateurs.

Je réutilise les interfaces existantes et à l'interface « gérer patient » je vais modifier les composants (voir Annexe). Hors cela, je vais rajouter des onglets (les activités contenant dans un parcours) et champs (les commentaires sur chaque activité) pour l'interface « dossier parcours » et rajouter un bouton de validation pour chaque activité.

4.3 Interface logiciel/ logiciel

Concernant les interfaces logiciel / logiciel, notre application web sera connectée à une base de données pour pouvoir sauvegarder et récupérer les données nécessaires au fonctionnement de l'application.

5 Description des fonctionnalités

Dans ce chapitre, je vais montrer les fonctionnalités de l'ancienne application (la description détaillée voir Annexe) et en plus présenter les nouvelles fonctionnalités que je vais implémenter. Voici le diagramme des cas d'utilisation **Figure 2**.

Les fonctionnalités en vert sont déjà présentes dans le système et elles fonctionnent sans problème.

Les fonctionnalités en orange sont présentes dans le système mais elles ne fonctionnent pas bien, il existe des bugs à corriger ou encore à améliorer. Le reste est les fonctionnalités à développer ou à corriger.

5.1 Revoir la suppression de la base de données

Du fait du problème de la suppression des éléments pendant l'utilisation de l'application, je dois restructurer une partie de la base de données pour résoudre ce genre de problème.

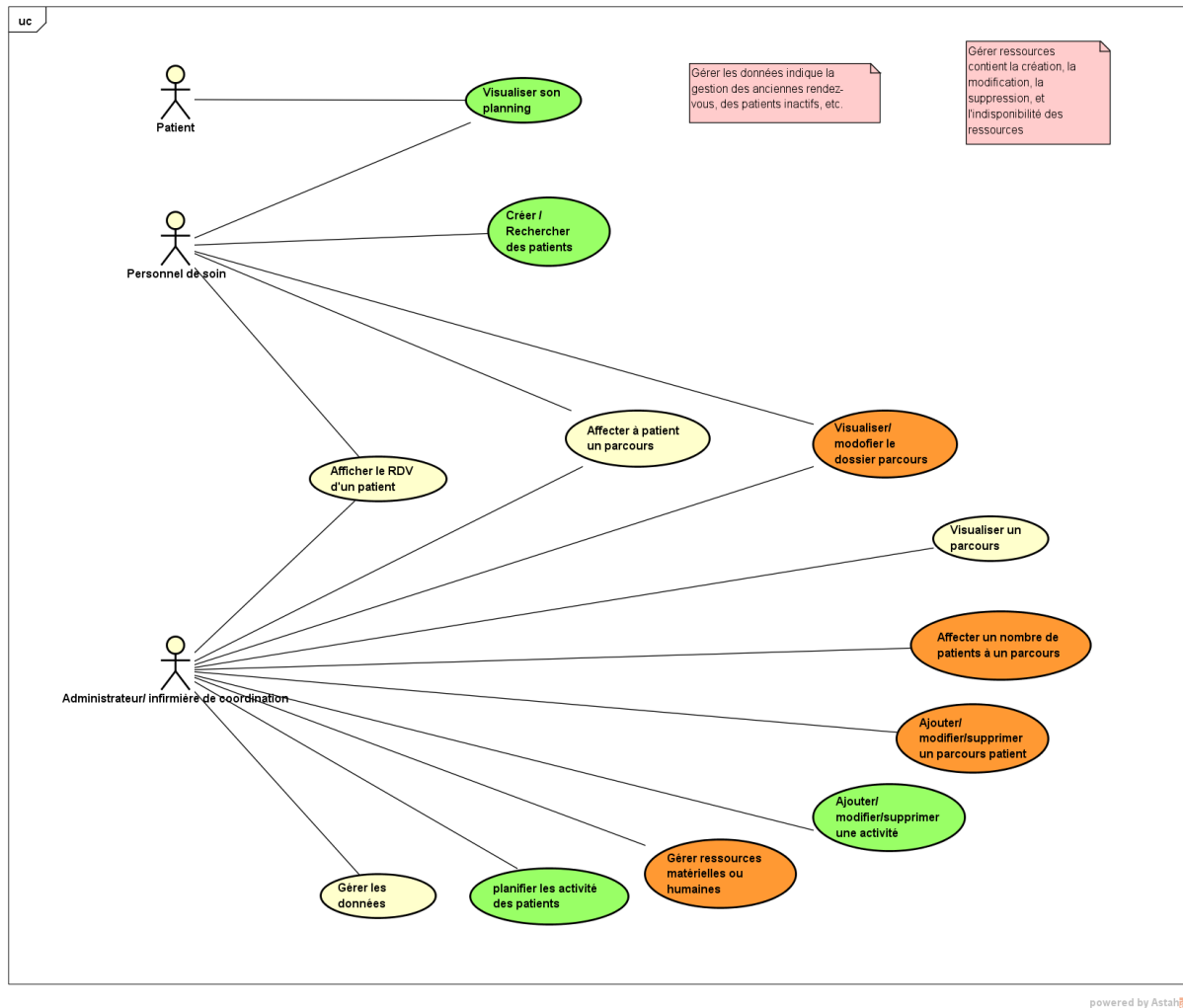


Figure 2 – Diagramme des cas d'utilisation

Dans le diagramme des cas d'utilisation, ça sera le cas "gérer les données".

En entrée, les tables « patient », « ressource », « parcours », « dossierparcours », « personnel » et « activité » dans la base de données seront utilisées. Et en sortie, une colonne qui indique l'état de cet élément (il est encore actif ou pas) sera ajouté en queue des tables.

5.2 Créer une fonctionnalité PURGE

Je vais ajouter une fonctionnalité PURGE pour la suppression des informations inutiles qui ralenti le fonctionnement de système.

En entrée, les tables « patient » et « dossierparcours » seront utilisées. Pour la sortie, la suppression des enregistrements dans ces tables aura lieu.

5.3 Améliorer la gestion des patients

Dans l'ancien projet, les fonctionnalités que nous pouvons effectuer sur un patient sont : Nouveau dossier, Afficher séjour et Dossiers parcours. Ces fonctionnalités ne sont pas assez claires pour l'utilisateur, et je vais les modifier comme les fonctionnalités suivantes : Modifier

patient, Gérer RDV (nouveau RDV ou modifier RDV), Afficher RDV (planning), Consulter les dossiers parcours et Supprimer RDV. En entrée, la page « recherche d'un patient » et les tables « patient », « parcours », « dossierparcours » seront utilisées. En sortie, les différentes pages liées à ces fonctionnalités seront créées. cette fonctionnalité correspond aux cas "afficher RDV d'un patient", "Affecter à patient un parcours", "visualiser/modifier dossier parcours" dans le diagramme des cas d'utilisation.

5.4 Améliorer l'affectation d'un parcours

Cette fonctionnalité est pour améliorer les étapes pour affecter un parcours au patient.

En entrée, la page « Afficher patient » et « Modifier Patient » seront utilisée. En sortie, dans la page « Modifier patient » il y aura un nouveau calendrier pour choisir la date de RDV, dans ce calendrier la case de date doit être coloré pour indiquer le ratio d'occupation de ce parcours. Il correspond au cas "affecter à patient un parcours" dans le diagramme des cas d'utilisation.

5.5 Faire valider l'activité

Chaque fois après une activité de parcours a été faite, il faut que l'administrateur vérifier cette activité pour que le patient puisse suivre l'avancement dans son parcours. Pour réaliser cette fonctionnalité, je vais ajouter un bouton de validation.

5.6 Refaire la partie de consultation sur les dossiers de parcours

Dans cette partie, le dossier de parcours sera présenté par la date de RDV. Pour chaque parcours, il y aura des onglets signifiant les activités contient dans ce parcours et pour chaque onglet il y aura des champs pour faire des commentaires. Dans le diagramme des cas d'utilisation ça sera le cas "visualiser/modifier le dossier parcours".

En entrée la page « dossier parcours » et les tables « parcours », « onglet », « champ » seront utilisées. Pour la sortie, les onglets et champs seront présents dans la page « dossier parcours ».

5.7 Faire le plan de parcours correspondre au objectif de nombre de patients

Dans la partie « gérer plan parcours », on peut donner un nombre de patient comme planning, hors cela, l'objectif de nombre de patients (patients attendus par jour) peut être aussi défini dans la partie « parcours » quand on modifie un parcours. Mais ces deux parties ne sont pas cohérentes, ça veut dire si on change un objectif dans « parcours », ce dernier ne changera pas dans la partie « plan parcours », donc il faut refaire cette partie pour que cette fonctionnalité peut fonctionner correctement.

En entrée, la classe « planparcours » et la classe « parcours » seront utilisée.

6 Structure générale de système

La structure du logiciel est basée sur l'architecture MVC (modèle vue contrôleur). Le modèle représente toutes les données de l'application (patient, ressources humaines, ressources matérielles, parcours, activités, etc.). Le contrôleur est utilisé pour faire le lien entre les données

et les différentes vues. Ce dernier permet également de gérer la cohérence et l'intégrité des données. Les vues représentent les pages web de l'application. Les tables de la base de données sont identiques aux projets réalisés précédemment mais certaines colonnes des tables seront modifiées (voir **Figure 1** (Annexe D)).

La fonctionnalité « suppression » et « PURGE » s'appuient sur les classes « patient », « ressource », « parcours », « dossierparcours », « personnel » et « activité » dans l'architecture dessus. La fonctionnalité « amélioration de gestion des patients » sera développée en utilisant les classes « dossierparcours », « parcours » et « patient ».

Pour la fonctionnalité « validation d'une activité » et la fonctionnalité « consultation de dossier parcours », les tables « onglet », « champ », « dossierparcours », « parcours », « patient », « dossiergenerique » et « activité » sont utilisées. En plus, pour la correspondance entre le plan parcours (le nombre de ce parcours pouvant être effectué dans un jour) et l'objectif de patients (le nombre de patients pouvant être affectés au certain parcours), on utilise les tables « planparcours » et « parcours ».

3

Etat de l'art

1 Logiciels existants

Sur le marché, il y existe des logiciels pour la gestion des patients, ou pour l'optimisation des ressources ou pour la planification des chimiothérapies, mais ils sont tous payants. Il y a peu de documentation disponible, et j'ai trouvé les fonctions qu'ils annoncent sur leur site comme suivant.

La société Effigen développe des solutions pour l'optimisation des hôpitaux de jours. Cette optimisation permet de réduire les délais de prise en charge et d'augmenter la productivité.

Leur solution se découpe en plusieurs logiciels, un qui permet de visualiser la planification des parcours patients mais également une solution permettant de reprogrammer les plannings en fonction des flux observés (retard, absence de patient, etc.).

Leur solution permettrait d'augmenter de 40 à 65% le taux de prise en charge des patients et de lisser la charge de travail au cours de la journée et de la semaine.

Gesplic est un logiciel de gestion et de planification des chimiothérapies. Ce dernier permet de suivre et de planifier les différentes séances de chimiothérapies d'un patient. Le logiciel calcule en temps réel les prochaines séances de chimiothérapie en fonction de l'analyse des précédentes.

IC-Plan est un logiciel permettant de gérer les patients dans un hôpital (gestion des lits d'hôpitaux suivie des séjours, etc.). Il existe également plusieurs logiciels de gestion de données sur les patients dans les hôpitaux et de gestion des personnels.

Hors cela, l'application créée par les étudiants dans leur PRD de Polytech Tours permet de planifier les différentes activités d'un parcours patient sous les contraintes des activités précédentes. Cette application est aussi pour la gestion des patients, des ressources et des parcours, en plus le patient et le personnel peuvent également visualiser leur planning. Notre projet est basé sur cette application existante.

2 Veille technologique

Pour l'affichage et la modification des plannings, j'ai besoin de trouver une librairie qui nous permet de faire cela. Dans notre cas nous avons besoin d'une librairie qui puisse afficher le calendrier sous forme de semaine, qu'on puisse effectuer des drags and drop d'événements.

Cette partie a été développée par Pochet Guillaume dans son PRD, donc je ne vais pas modifier cette partie et je vais continuer à utiliser le FullCalendar pour la modification des plannings.

4

Analyse et conception

1 Conception des fonctionnalités

Dans ce chapitre, je vais analyser en détail les fonctionnalités que je vais implémenter dans ce projet. La liste des fonctionnalités à faire est :

- ★ Revoir la suppression de la base de données
- ★ Créer une fonctionnalité PURGE
- ★ Améliorer la gestion des patients
- ★ Améliorer l'affectation d'un parcours
- ★ Faire valider l'activité
- ★ Refaire la partie de consultation sur les dossiers de parcours
- ★ Faire le plan de parcours correspondre au objectif de nombre de patients

2 Analyse sur les fonctionnalités

2.1 Revoir la suppression de la base de données

Avec l'ancienne application, la plate-forme nous ne permet pas à supprimer les choses correctement à cause de la cascade des différentes classes, encore après la suppression des autre choses lié avec l'élément supprimé ne sont pas traitées correctement. D'abord la suppression doivent être réalisée sans erreur de la base de données, ensuite il faut vérifier qu'après la suppression, les fonctionnalités d'application ne sont pas affectées. Par exemple, si on supprime un patient, tous les dossiers parcours associés à ce patient doivent aussi être supprimé, et la planification de ce patient doit disparaître.

2.2 Créer une fonctionnalité PURGE

Pour les patients qui sont plus actifs dans le système (par exemple, les patients sans RDV pour plus de 3 ans), et les RDV qui ont été passés longtemps, on ne garde plus ces informations car la

base de données sera très importante si on garde toutes les informations et en même temps elle deviendra lente.

Je vais ajouter une fonctionnalité PURGE pour la suppression des informations inutiles, cela va faciliter la gestion de système et diminuer la charge de la base de données.

2.3 Améliorer la gestion des patients

La liste de travaux que nous pouvons effectuer sur un patient a été implémentée mais elle est mal construite et correspond pas à notre besoins, donc cette partie sera refaite dans mon projet.

Dans l'ancienne application, les fonctionnalités sur les patients sont suivantes : Nouveau parcours, Dossier parcours, et Afficher séjour. Ces fonctionnalités n'effectuent pas vraiment ce qui est décrit sur l'application. Par exemple, "Dossier parcours" n'est pas réalisé et "Nouveau parcours" nous ne permet q'ajouter un nouveau parcours à un patient. On souhaite d'avoir plus d'opération sur un patient et plus logiquement. Donc pour mon projet je vais les changer comme : Modifier patient, Gérer RDV (ici on peut choisir à créer un nouveau parcours ou seulement modifier un parcours existant et Supprimer un RDV); Afficher RDV (ici on peut sélectionner la date de RDV et voir le planning); Consulter les dossiers parcours (à gauche seront les dates de RDV, on peut consulter les parcours selon la date de RDV; pour chaque parcours les activités seront présentées comme les onglets et pour chaque onglet, on a des champs pour mettre les commentaires). L'interface sur laquelle ces modifications seront effectuées est : **Figure 1** (Annexe A)

- ★ Précondition :
 - L'utilisateur doit avoir recherché un patient par son nom ou prénom
 - L'utilisateur doit posséder un compte administrateur et être connecté
- ★ Post-condition :
 - Si Modifier patient, l'utilisateur sera ramené à l'interface de modification de patient
 - Si gérer RDV, l'utilisateur sera ramené à une nouvelle page de gestion des RDVs dont ajout/ modification/ suppression d'un RDV sont tous sur cette page.
 - Si afficher RDV, l'utilisateur sera ramené à l'interface de planning
 - Si consulter les dossiers parcours, l'utilisateur sera ramené à l'interface de dossiers parcours

2.4 Améliorer l'affectation d'un parcours

Lorsqu'on modifie ou crée un parcours d'un patient, il faut choisir une date quand le patient sera disponible. Dans l'application existante, les dates doivent être présentées par des cellules avec la couleur indiquant la capacité de ce jour, mais ce n'est pas réalisé. Donc dans mon PRD, je dois compléter cette fonctionnalité. La capacité d'un parcours dans une journée est présentée par le ratio d'occupation des dossiers parcours existant de ce jour.

2.5 Faire valider l'activité

Chaque fois après une activité de parcours a été faite, il faut que l'administrateur vérifie cette activité pour que le patient puisse suivre l'avancement dans son parcours. Pour réaliser cette fonctionnalité, je vais ajouter un bouton de validation. L'interface sur laquelle ces modifications seront effectuées est : **Figure 1** (Annexe B)

- ★ Précondition :

- Le patient doit avoir fini une activité
- Le personnel ou l'administrateur doit avoir fait un commentaire s'il y en a besoin
- ★ Post-condition :
 - Dans le planning, le patient sera avancé

2.6 Refaire la partie de consultation sur les dossiers de parcours

Dans cette partie, le dossier de parcours sera présenté par la date de RDV. Pour chaque parcours, il y aura des onglets signifiant les activités contient dans ce parcours et pour chaque onglet il y aura des champs pour faire des commentaires. Pour chaque onglet, il y aura des champs défauts et on peut ajouter nouveau champ quand on a besoin.

L'interface sur laquelle ces modifications seront effectuées est : **Figure 1** (Annexe B)

- ★ Précondition :
 - L'utilisateur doit avoir sélectionné Consulter les dossiers parcours
- ★ Post-condition :
 - L'utilisateur sera amené à l'interface de dossier de parcours
 - Les onglets représentant les activités dans un parcours appartiendront
 - Les champs pour mettre les commentaires appartiendront

2.7 Faire le plan de parcours correspondre au objectif de nombre de patients

Dans la partie « gérer plan parcours », on peut donner un nombre de patient comme planning, hors cela, l'objectif de nombre de patients (patients attendus par jour) peut être aussi défini dans la partie « parcours » quand on modifie un parcours. Quand on change l'objectif de patient dans la page « parcours », le nombre de patient dans « planparcours » doit être changé au même temps.

5

Bilan et conclusion

1 Tâches réalisées

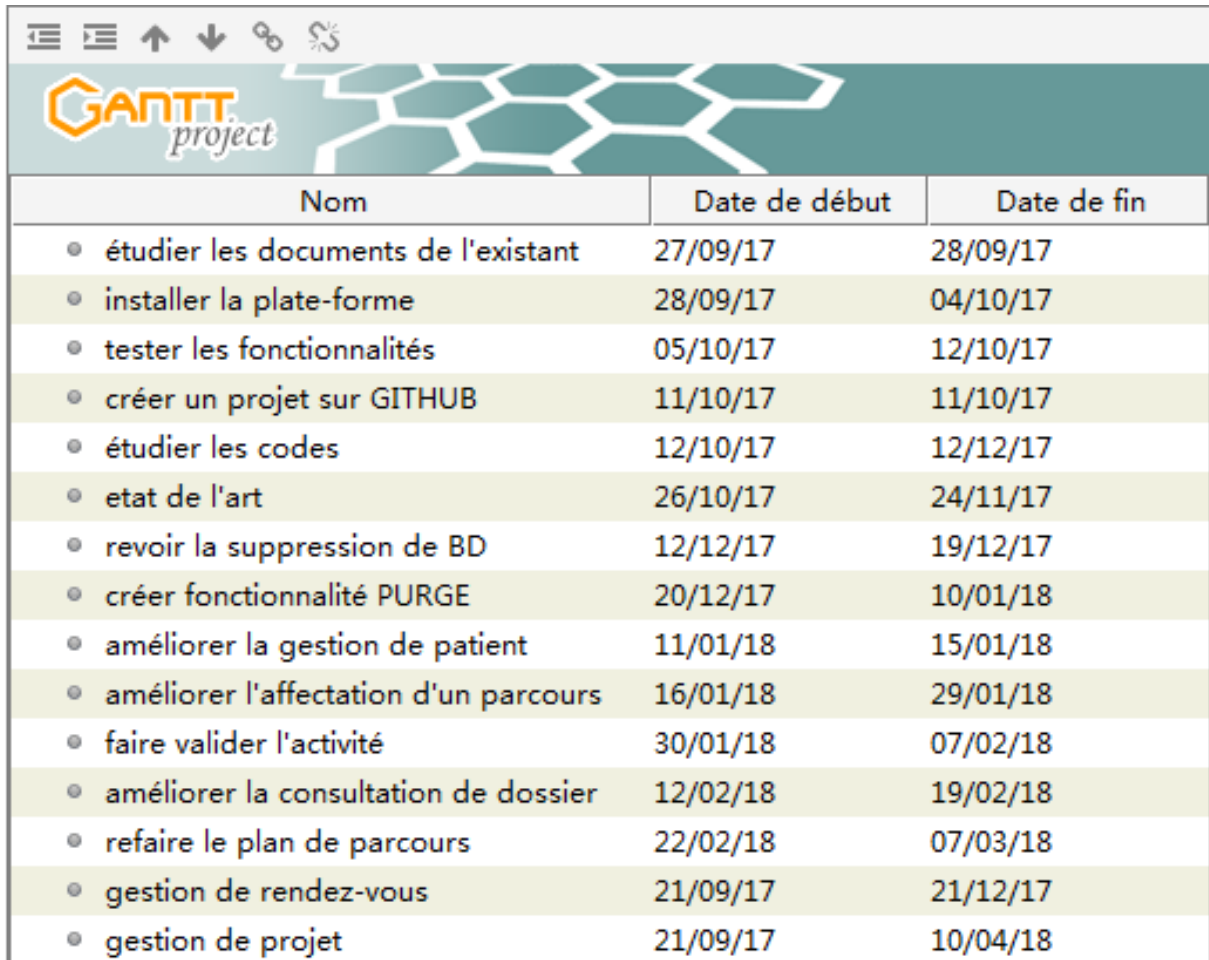
Les différentes tâches que j'ai déjà réalisées sont :

- la mise en place des outils de gestion de projet (Git, trello)
- la mise en place du projet existant (l'installation de la plate-forme etc.)
- les spécifications
- l'état de l'art et la veille technologique
- tests sur les fonctionnalités existantes

2 Plan de développement

Voici la liste des tâches de notre projet **Figure 1**.

Pour la réalisation de projet, je vais travailler avec une méthode agile qui a un cycle de vie cascade, ça veut dire que chaque fois quand je finis une tâche je dois la valider avec mon tuteur. Je dois prendre des rendez-vous régulièrement avec mon tuteur académique et avec le client s'il y en a besoin afin d'avoir des retours réguliers sur le projet et sur les attentes du client.



The image shows a screenshot of a Gantt project management tool. At the top, there is a toolbar with icons for list view, Gantt view, zoom in, zoom out, link, and refresh. Below the toolbar is the 'GANTT project' logo. The main area contains a table with three columns: 'Nom', 'Date de début', and 'Date de fin'. The table lists 15 tasks, each preceded by a bullet point. The tasks are: étudier les documents de l'existant, installer la plate-forme, tester les fonctionnalités, créer un projet sur GITHUB, étudier les codes, etat de l'art, revoir la suppression de BD, créer fonctionnalité PURGE, améliorer la gestion de patient, améliorer l'affectation d'un parcours, faire valider l'activité, améliorer la consultation de dossier, refaire le plan de parcours, gestion de rendez-vous, and gestion de projet. The dates range from 27/09/17 to 10/04/18.

Nom	Date de début	Date de fin
• étudier les documents de l'existant	27/09/17	28/09/17
• installer la plate-forme	28/09/17	04/10/17
• tester les fonctionnalités	05/10/17	12/10/17
• créer un projet sur GITHUB	11/10/17	11/10/17
• étudier les codes	12/10/17	12/12/17
• etat de l'art	26/10/17	24/11/17
• revoir la suppression de BD	12/12/17	19/12/17
• créer fonctionnalité PURGE	20/12/17	10/01/18
• améliorer la gestion de patient	11/01/18	15/01/18
• améliorer l'affectation d'un parcours	16/01/18	29/01/18
• faire valider l'activité	30/01/18	07/02/18
• améliorer la consultation de dossier	12/02/18	19/02/18
• refaire le plan de parcours	22/02/18	07/03/18
• gestion de rendez-vous	21/09/17	21/12/17
• gestion de projet	21/09/17	10/04/18

Figure 1 – Liste des tâches de notre projet

Deuxième partie

Développement

6

Méthodologie de suivi et gestion de projet

1 Suivi par l'encadrant

Concernant la méthodologie de suivi de gestion de projet, j'ai pris des rendez-vous avec mon encadrant environ toutes les 2-3 semaines et faire un point sur ce qui a été fait par une démonstration de l'application. Après chaque rendez-vous, nous redéfinissons les travaux à continuer pour la suite de projet.

2 Versionning et gestion de projet

Pour la versionning du projet, j'ai créé un référentiel git chaque fois quand une grande fonctionnalité est finie complètement je fais la mise à jour des codes **Figure 1**.

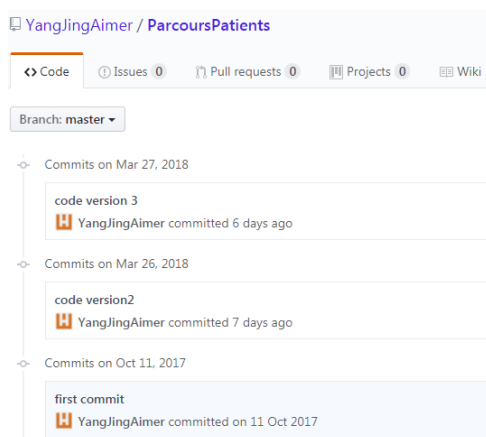


Figure 1 – Versionning du projet

7

Mise en oeuvre

1 Fonctionnement général

Après la recherche d'un patient, il faudra permettre de modifier les informations de patient, de permettre de gérer des RDVs de ce patient et en plus de consulter les dossiers d'un patient. Quand on supprime un élément, il faudra vérifier que d'autres éléments associés à cet élément ne seront pas affectés ou qu'il seront aussi traités correctement pour éviter des problèmes (par exemple, le problème de planification).

2 Conception de la suppression

Concernant la conception de la suppression, j'ai d'abord vérifié que les requêtes sql fonctionnent et ensuite pour chaque élément, vérifier que les autres éléments associés ne posent pas d'erreur. Par exemple, si on supprime un personnel, il faut vérifier que toutes les activités affectées à ce personnel seront remettre dans la liste à planifier.

2.1 Principe

Le principe est assez simple. Pour cela, d'abord j'ai modifié les relations des tables en "delete/upadte en cascade", après j'ai modifié les codes pour que les actions d'après la suppression correspondent aux notre besoins.

2.2 limite

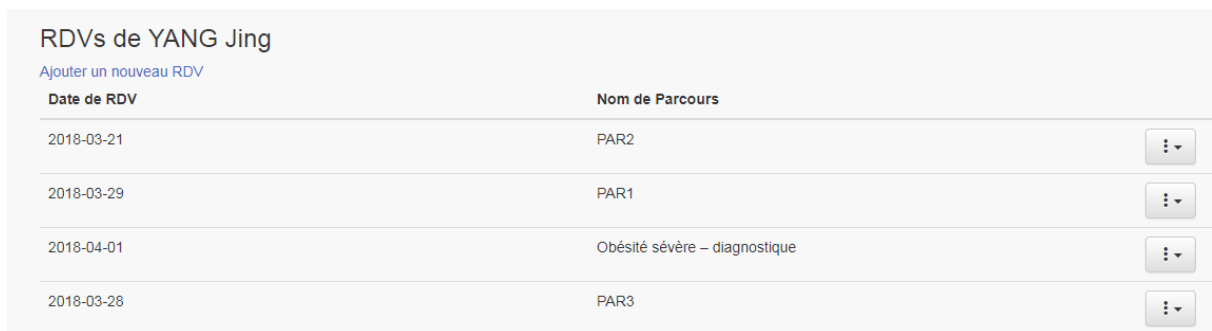
Les actions que nous traitons après la suppression peuvent être pas complètes car c'est possible que nous négligeons des relations entre deux éléments.

3 Conception de gérer RDV

Concernant la gestion de RDV, j'ai créé une nouvelle page. Il faudra permettre d'ajouter un nouveau RDV et de modifier un RDV à un autre parcours dans une autre date. Dans la nouvelle page, il faut afficher tous les RDVs avec la date et nom.

3.1 Principe

Le principe est d'utiliser l'identifiant de patient passé sur la page de recherche et afficher tous les dossiers associés à ce patient puis lister les résultats **Figure 1**.



RDVs de YANG Jing	
Ajouter un nouveau RDV	
Date de RDV	Nom de Parcours
2018-03-21	PAR2
2018-03-29	PAR1
2018-04-01	Obésité sévère – diagnostique
2018-03-28	PAR3

Figure 1 – La gestion des RDVs

4 Conception de consulter dossiers parcours

Concernant la consultation des dossiers, il faudra que chaque dossier parcours contient des onglets qui signifie une activité dans le parcours et des champs permettant de faire des commentaires.

4.1 Principe

Le principe est d'utiliser l'identifiant de patient pour trouver tous les dossiers parcours de ce patient et ensuite trouver les onglets et les champs selon la table <constituerdossier>. Pour réaliser la bonne consultation, j'ai d'abord réalisé le lien entre la table activité et onglets pour que chaque onglet correspond à une activité dans un parcours. Ensuite, j'ai réalisé la constitution de dossier entre la table <dossierparcours>, <onglet> et <champ> donc chaque fois quand on crée un nouveau dossier, on ajoute dans la table <constituerdossier> un nouveau enregistrement **Figure 2**.

Dossier parcours de YANG Jing

2018-03-21
2018-03-29
2018-04-01
2018-03-28

RDV paramédical ECG Bilan biologique Echo hépatique Calorimétrie Entretien psy Entretien infirmier Entretien diet

synthèse Collation

Observation

Date d'observation

new OK

+ Ajouter un nouveau champ pour l'onglet

✓ Sauvegarder

Figure 2 – La consultation des dossiers

8

Validation et tests

1 Tests unitaires

Mon projet consistait principalement à développer des pages web afin de faire fonctionner la gestion des patients, des dossiers parcours et des ressources. Le projet étant principalement graphique, je n'ai pas mis en place de tests unitaires. Pour vérifier les fonctionnalités, j'ai effectué des données de tests afin de pouvoir tester la gestion des différents éléments, et la suppression des différents éléments.

2 Tests fonctionnels

Pour tester les suppressions des éléments, j'ai effectué un jeu de données de tests qui contient 5 activités, 5 personnels, 5 ressources maternelles, 5 parcours, et 5 patients à créer/modifier/-supprimer. Mes données contiennent aussi 5 RDVs à ajouter/ modifier.

Ces données m'ont permis de tester toutes les contraintes suivantes :

- Toutes les suppressions s'effectuent correctement
- Les informations sur un patient peuvent être modifiées indépendamment
- Gestion de RDV fonctionne (y contient l'ajout, la modification, et la suppression)
- Les dossiers de parcours peuvent être consultés avec des onglets corrects
- Les dossiers de parcours devront être affichés par les dates de RDV
- Le calendrier affiche la couleur pour indiquer le ratio d'occupation d'un parcours
- Le plan de parcours peut être modifié

Tous les détails de tests, les résultats attendus et les résultats obtenus sont notés dans le rapport de test (Voir annexe).

9

Avancement du projet

Dans cette partie, nous allons faire un point sur l'avancement du projet.

1 Fonctionnalités réalisées

Voici une liste des fonctionnalités qui ont été réalisées :

- La gestion des activités (ajout/modification/suppression)
- La gestion des ressources (ajout/modification/suppression, personnelle/matérielle)
- La gestion des patient (ajout/modification/suppression)
- La gestion de RDVs d'un patient (ajout/modification/suppression)
- Consultation des dossiers parcours d'un patient (affichage de RDVs)

2 Fonctionnalités reste à faire

Voici la liste complète des fonctionnalités restantes qui ont été définis au travers de plusieurs rendez vous :

- La validation d'une activité
- Planification en temps réel
- Planification automatique
- Ajouter des indisponibilités de ressources (personnelle et matérielle)

10

Re-productivité

1 Pré-requis et installation

Le projet a été développé à l'aide de l'IDE netbeans, compatible avec Windows, Mac OS et Linux. La base de données utilisée est une base de données MySQL. L'IDE est disponible en téléchargement sur le site officiel de netbeans (<https://netbeans.org/>).

Pour pouvoir utiliser l'application, un serveur web et de base de données sont nécessaires.

Pour ma part, j'ai utilisé WAMP (<http://www.wampserver.com/>).

2 Déploiement

Pour déployer l'application, il suffit de réaliser ces différentes étapes :

- Copier toutes les sources dans un dossier www (WAMP par exemple)
- Exécuter le script « subway.sql » dans PhpMyAdmin afin de créer et d'ajouter des données à la base de données
- Aller sur l'adresse suivante : <http://localhost/subway/>.

3 Documentation

Toute la documentation du projet a été générée par doxygen. La documentation est sous le répertoire documentation de l'application.

Chaque classe a été documentée suivant ce modèle : (Voir [Figure 1](#))

Chaque méthode a été documentée suivant ce modèle : (Voir [Figure 2](#))

Il y a un guide développeur afin d'aider les personnes qui reprendront ce projet (voir annexe).

```

/**
 * \file      M_RessourcesMat.php
 * \author    Guillaume Pochet
 * \version   1.0
 * \date      09 Mars 2017
 * \brief     Définit les méthodes liées aux ressources matérielles
 *
 * \details   Ce fichier permet de définir les méthodes de gestion des ressources
 */
class M_RessourcesMat extends CI_Model {

```

Figure 1 – Documentation d’une classe

```

/**
 * \brief     Récupère toute la liste des ressources matérielles
 * \details   Récupère toute la liste des ressources matérielles
 * \param     Aucun
 */
public function getAllRessourcesMat() {

```

Figure 2 – Documentation d’une méthode

11

Conclusion

Cette application de gestion de parcours patient maintenant permet de gérer toutes les données (parcours, personnel, ressource matérielle, activité et patient). Elle permet de prendre un RDV ou modifier des RDV pris précédemment, et elle permet d'afficher les dossiers parcours d'un patients dont on peut laisser des commentaires sur un champ. De plus, elle permet de planifier des activités manuellement.

Il reste encore des fonctionnalités à compléter par exemple la planification en temps réel.

Ce projet est un bon expérience dans le programmation web pour moi, de plus il m'apporté des opportunités de pratiquer le langage SQL. Pendant ce projet, mon encadrant m'a donnée beaucoup de conseils sur les tests, ça m'a aidé de penser plus strictement.

Annexes

A

Maquettes détaillées

Interface Recherche Patient

[←](#)
[→](#)
[↻](#)

[Accueil](#)
[Ressources](#)
[Activités](#)
[Parcours](#)
[Patients](#)
[Planning](#)
[Se déconnecter](#)

Rechercher d'un patient

Résultats de recherche

▼ Nom	▼ Prénom	▼ Date de naissance	▼ Numéro SS	
Laurent	Bernard	1993-09-12	12345	▼
Jean	Paillard	1995-10-29	12367	▼
Matt	Damon	1978-2-18	23789	▼

Gérer RDV
Afficher RDV
Consulter dossiers Parcours
Supprimer RDV

Figure 1 – Gestion de patient

Interface Dossiers parcours

← → ↻ <http://www.climab.fr/dossierParcours>

Accueil Ressources Activités Parcours Patients Planning Se déconnecter

Dossiers parcours de Laurent Bernard (date des RDVs) Sauvegarder

	activité 1	activité 2	activité 3	activité 4	activité 5
2017-10-11 champ 1					
2017-10-26 champ 2					
champ 3					

+ Ajouter un nouveau champ Valider activité

Figure 1 – Valider activité et consulter les dossiers parcours

C

Les fonctionnalités déjà implémentées

L'application web a été produite par les étudiants au cours de l'année 2015-2016 de leur projet collectif et deux étudiants de son PRD. Ils ont développé une plate-forme permettant les fonctionnalités suivantes : gérer les personnels, gérer les ressources matérielles, gérer les activités, gérer les parcours et gérer le plan de parcours. Hors cela, la plate-forme peut aussi créer et rechercher des patients, puis leur affecter des parcours et planifier les activités.

1 Gérer les personnels

Chaque personnel a une fonctionnalité dans l'hôpital, l'administrateur peut les modifier, supprimer et les mettre en indisponibilité.

- Modifier un personnel
- Supprimer un personnel
- Mettre un personnel en indisponibilité

2 Gérer les ressources matérielles

Chaque ressource matérielle a un nom et une spécialité.

- Modifier une ressource matérielle Cette fonctionnalité est implémenté mais elle ne fonctionne pas bien car elle pose le problème de la base de données, il faut une amélioration.
- Supprimer une ressource matérielle Cette fonctionnalité a été implémentée mais elle ne fonctionne pas donc on a besoin de la faire fonctionner.
- Mettre un personnel en indisponibilité

3 Gérer les activités

Chaque activité doit avoir un nom et une durée, elle sera faite dans une certaine salle et par un type de personnel.

- Ajouter une nouvelle activité
- Modifier les activités existantes
- Supprimer les activités qui n'existent plus

4 Gérer les parcours

Un parcours se compose par plusieurs activités dont il y a un ordre entre tous les activités, donc il faut faire attention aux activités qui doivent être en précédentes des autres.

- Ajouter un nouveau parcours
- Modifier un parcours
- Supprimer un parcours
- Visualiser un parcours

5 Gérer le plan des parcours

Dans un hôpital de jour, la capacité de recueillir les patients est limite, donc dans une journée il peut affecter un certain nombre de patients au même parcours.

6 Visualiser le planning (patient)

Une fonctionnalité « afficher planning du patient » est disponible et permet de visualiser l'heure de début et de fin de la prochaine activité du patient dans son parcours. Cette visualisation est visible par l'administrateur, le personnel de soins et également par le patient.

7 Affecter patient à un parcours

Le patient peut être affecté à un parcours en précisant les disponibilités de ce dernier. Les disponibilités du patient seront demandées sous formes d'horaires (exemple : 8h-17h). Cette fonctionnalité est disponible à l'administrateur et aux personnels de soins.

L'administrateur ou infirmière de coordination peuvent affecter un patient à un parcours pour une journée donnée et pouvoir visualiser également le nombre de patients déjà affecté sur cette dernière.

Lors d'un clic sur le bouton affecter, l'administrateur sera redirigé vers une page où il pourra choisir le patient à affecter pour la date choisie précédemment

8 Visualiser et modifier le planning de toutes les ressources

L'administrateur peut visualiser et modifier en temps réel le planning de toutes les ressources et également d'une ressource en particulier. La modification peut s'effectuer de façon manuelle (en déplaçant les activités). L'administrateur doit pouvoir planifier automatiquement le planning de ses ressources pour une journée donnée. La planification automatique ne sera pas incluse dans le cadre de mon projet de recherche et développement. Ces fonctionnalités ne sont disponibles qu'à l'administrateur. Les différentes interfaces ont été étudié dans la partie veille technologique sur fullCalendar.

Modèle conceptuel des données

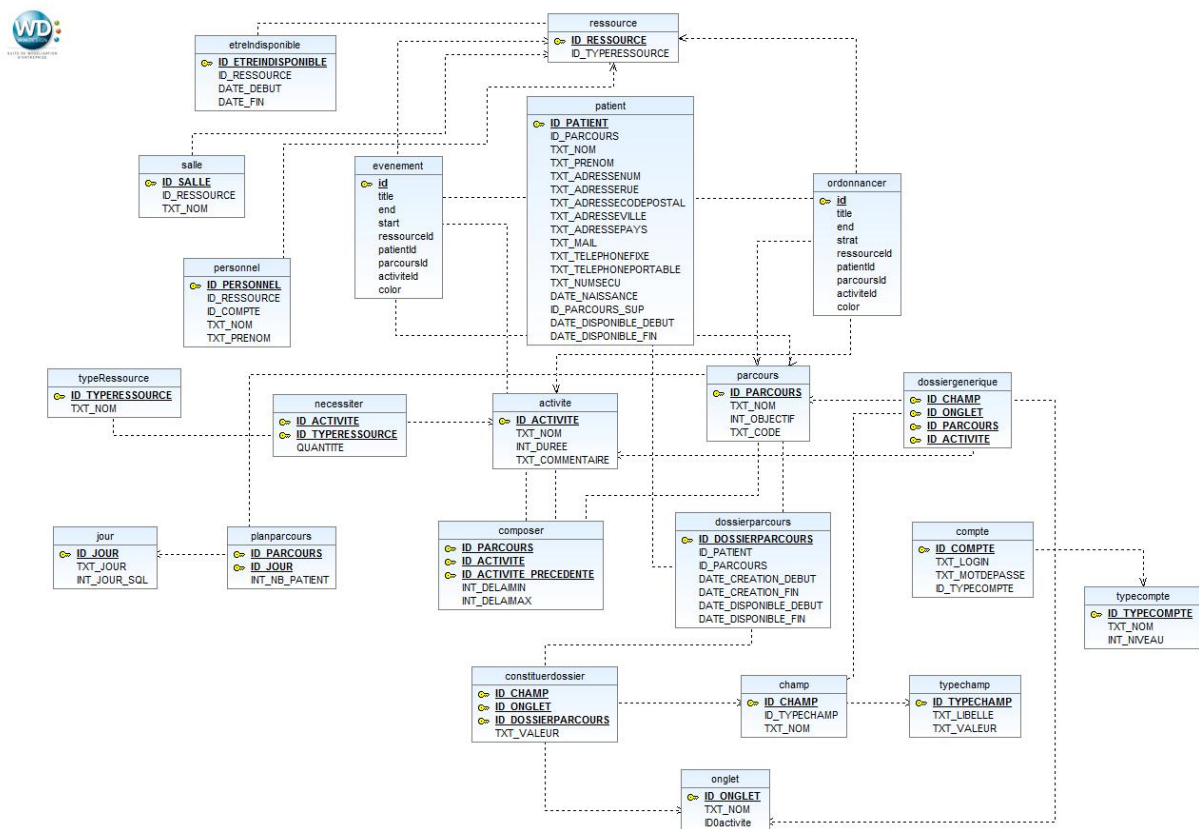


Figure 1 – *Diagramme de classes*

Notre Système d'Information est basé sur 22 tables qui ont été créées par les étudiants précédents [2], pour mon projet je n'ajouterai pas de nouvelle table. Nous allons maintenant faire un petit détail de l'ensemble des tables afin de connaître leur utilité.

E

Explications des tables de notre modèles

Voici l'explication sur les tables créées dans l'ancien projet [2] que je vais continuer à utiliser.

Activité : Table regroupant l'ensemble des informations concernant une activité. C'est cet élément qui constitue les parcours. Une activité peut obliger la réalisation de certaines autres activités avant d'être réalisable. Cette notion de dépendance dépend du parcours en cours. Cette notion de précedence est sera vue plus loin via la table composer. En plus de cela, une activité a besoin de ressources. Ce lien se fait par la table necessiter.

Champ : Table contenant l'ensemble des champs qu'il est possible d'ajouter dans un onglet d'un dossier Parcours. A chaque champ est lié un type de champ, que nous verrons plus loin, avec la table typechamp.

Composer : Table permettant de faire le lien entre un parcours et une activité. Par l'intermédiaire de cette table nous pouvons dire en fonction d'un parcours et d'une activité s'il y a des besoins en termes de précedence. Chaque ligne, dans cette table, a pour signification : « Pour l'activité A dans le parcours P, il faut avoir réalisé l'activité B avant, et ce dans un délai compris entre delaiMin et delaiMax. ». Il est important de noter qu'il est possible de mettre à « null », l'id de l'activité précédente.

Compte : Table regroupant l'ensemble des comptes qu'ils soient des comptes patient ou des comptes de type ressource médicale.

Constituerdossier : Table permettant de faire le lien entre un dossier parcours et les informations qui le constituent. En effet, nous retrouvons pour chaque dossier parcours et pour chaque onglet dans ce dernier, la valeur des champs le composant.

Dossiergenerique : Table définissant, pour un parcours, les onglets et les champs que tous les dossiers parcours doivent avoir impérativement.

Dossierparcours : Table renseignant les informations génériques d'un dossier parcours. Soit le patient associé, le parcours, dates de création et de dernière modification.

Etreindisponible : Table regroupant l'ensemble des indisponibilités pour une ressource. Cette indisponibilité est caractérisée par une date de début et de fin, acceptant toutes les deux le renseignement de l'heure.

Jour : Table qui contient les jours de semaine, ainsi que leurs index sous MySQL. Cette table a un enjeu au niveau des prévisions du nombre de patient par jour pour un parcours.

Necessiter : Table permettant de renseigner les types de ressources requises pour une activité, ainsi que la quantité nécessaire.

Onglet : Table des onglets disponibles pour constituer un dossier parcours.

Parcours : Table décrivant un parcours de façon générale.

Personnel : Table contenant l'ensemble de personnel médical de l'établissement. Chaque personnel a un compte, et est considéré comme une ressource.

Planparcours : Table regroupant l'ensemble des objectifs concernant le nombre de patients pour un parcours pour un jour donné.

Ressource : Table faisant le lien entre la table typeressource et personnel ou materiel. Ce lien sera expliqué plus en détail avec la table typeressource.

Typechamp : Table regroupant les différents type de champ qu'il est possible d'ajouter dans un dossier parcours. Elle contient également le code HTML des composants, permettant ainsi une mise en page en accord avec les autres éléments des pages.

Typecompte : Table utilisée pour la gestion des droits.

Typeressource : Table contenant les types de ressource d'un point de vue activité. En effet, une activité peut avoir besoin d'un type de ressource bien caractéristique (ex : IDE obésité). C'est pourquoi nous avons un double niveau de type de ressource. Un concernant les activités (typeressource) et un second plus d'un point de vue logique générale (personnel, materiel).

Ordonnancer : Table de fait de notre système. C'est la table la plus importante. Chaque ligne veut dire : « Pour le patient P qui fait le parcours Pa à la date D, il a besoin de la ressource R pour faire l'activité A de start à end. Cette table contient la planification réalisée de manière manuelle ou automatique (pas implémenté pour le moment).

Evenement : Table identique à la table Ordonnancer. Elle a le même but que la table ordonnancer mais cette table contient la planification en cours. Cette table permet de donc de pouvoir sauvegarder ou restaurer la planification en fonction des modifications que l'utilisateur a effectuées.

F

Gestion de projet

1 méthode de suivi de projet

Mon premier but de ce projet est de reprendre l'application existante et l'étudier profondément pour comprendre tous les fonctionnalités réalisées (recherche) et donc pouvoir corriger les bugs et ensuite faire des améliorations sur cet application (développement).

2 Outil de versionning

Pour mon PRD, j'ai décidé d'utiliser l'outil Git pour la versionning. C'est un bon choix pour la gestion de projet et il est facile à utiliser.

3 Diagramme de Gantt

Le diagramme de gantt a été renouvelé pour plusieurs raison.

- Les réalisations des fonctionnalités ont des contraintes, par exemple, la consultation devra après réalisé la gestion des RDVs, donc l'ordre des réalisation est plus flexible.
- Les suppressions a été réalisées donc la fonctionnalité PURGE n'aura pas besoin d'être réalisée.
- La fonctionnalité "valider l'activité" n'est pas réalisé car le temps n'est pas suffisant, nous préferions fournir un code fonctionnel plutôt que d'en faire un maximum

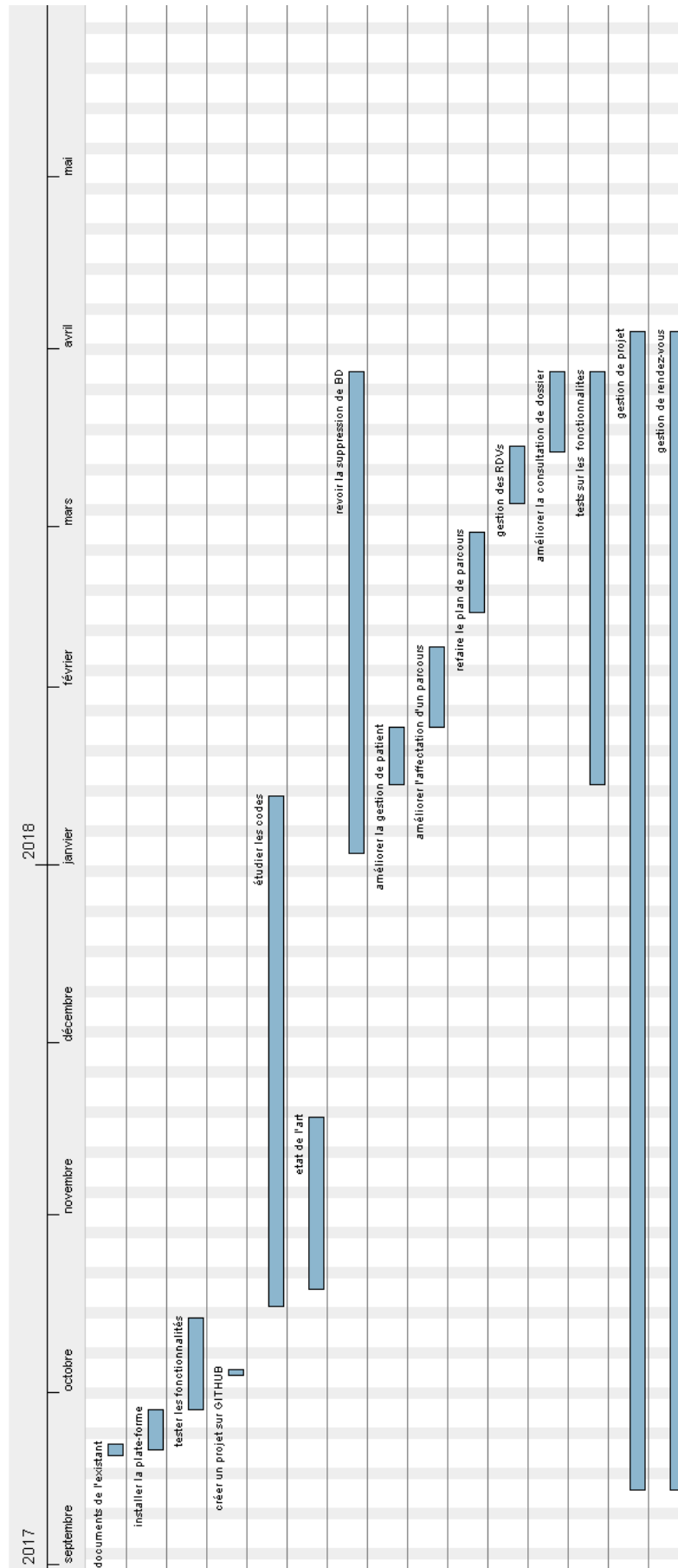


Figure 1 – Diagramme de Gantt

1 Structure des sources de l'application

L'application web a été développée en utilisant les technologies suivantes :

- PHP
- HTML, CSS
- Javascript

Pour le code PHP, nous avons utilisé un Framework PHP (codeIgniter) basé sur le modèle MVC (modèle vue contrôleur). L'utilité de ce Framework est de pouvoir séparer les données des différentes vues permettant l'affichage de ces données. Pour le CSS, nous avons également utilisé bootstrap afin d'avoir un rendu agréable sans avoir des capacités de graphistes.

Voici l'architecture générale de notre application web :

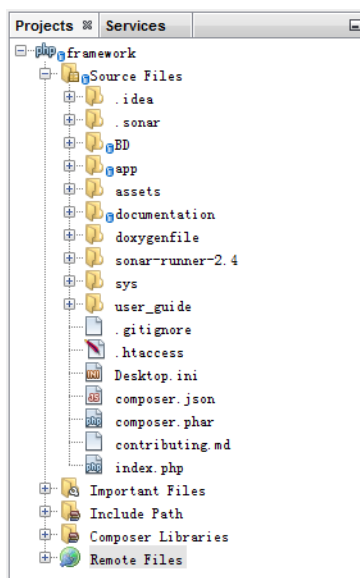


Figure 1 – La structure des sources d'application

Nous nous plaçons dans le répertoire source files qui est le répertoire qui nous intéresse le plus. Dans ce répertoire, nous trouvons un dossier BD qui contient le script à exécuter permettant

l'ajouter de données tests afin d'utiliser l'application.

Le répertoire « assets » contient tous les fichiers images, javascript, css nécessaire à notre application.

Le répertoire documentation contient toute la documentation nécessaire à la compréhension du code.

Les répertoires « sys » et « user_guide » ne nous intéressent pas forcément. Le premier contient tous les fichiers nécessaires à l'utilisation du Framework codeIgniter.

Le répertoire qui nous intéresse le plus est le répertoire « app » qui contient tous le code de notre application.

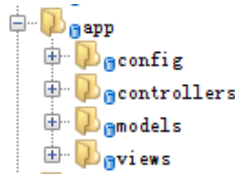


Figure 2 – Le répertoire app

Dans ce répertoire nous trouvons 4 sous répertoires :

- Config (répertoire de configuration du projet comme la base de données etc.)
- Controllers (tous les contrôleurs de notre application)
- Models (toutes les données de notre application : voir diagramme de classe)
- Views (les vues permettant l'affichage des données)

2 Contrôleurs

Tous les contrôleurs héritent d'une classe CI Controller (contrôleur de base du Framework codeIgniter). Les contrôleurs permettant de faire le lien entre les données et les vues. Les contrôleurs permettent dans la majeure partie des cas de faire appel à un modèle qui lui-même exécute une requête à la base de données et le contrôleur envoie ces données à la vue qui se charge de les afficher. Nous allons prendre un exemple pour voir l'utilisation d'un contrôleur.

```

/** \file      Activites ...10 lines */
class Activites extends CI_Controller {

    function __construct() {
        parent::__construct();
        if ($this->session->userdata("username") == null)
            redirect('/Auth', 'refresh');
        if ($this->session->userdata("level") == "1")
            redirect('/AffichageSejour', 'refresh');
    }

    /** \brief      Affichage la liste des différentes activités ...7 lines */
    public function index() {
        $this->load->model('M_Activite');
        $data = array();
        $data['activite'] = $this->M_Activite->getAllActivites();
        $data['chemin'] = '/activite/V_activite';
        $this->load->view('/V_generale', $data);
    }
}

```

Figure 3 – L'exemple de contrôleur

Dans l'exemple ci-dessus, nous prenons le cas du contrôleur « Activites » et la méthode « index » permettant d'afficher toutes les activités sur une page. La méthode charge le modèle nécessaire à la récupération des différentes activités, nous faisons appel à la méthode « getAllActivites » du modèle « M_Activite » permettant de récupérer toute la liste des activités. Ensuite, nous créons un tableau data permettant de récupérer le résultat de la méthode « getAllActivites » ainsi que

le chemin de la vue correspondante. Ensuite, notre contrôleur charge la vue principale (voir dans la partie explication des vues) en lui donnant également le chemin de la vue « V_activite » ainsi que les données provenant du modèle. Chaque contrôleur et chaque méthode sont basés sur cette structure.

3 Modèles

Dans cette partie, nous allons voir la structure des différentes modèles de notre application. Les modèles sont présents dans notre dossier « models ».

```
public function supprActivite($id) {
    $txt_sql = "DELETE FROM activite
              WHERE id_activite = " . $id;
    $query = $this->db->query($txt_sql);
}
```

Figure 4 – L'exemple de modèle

Les modèles exécutent uniquement des requêtes à la base de données.

4 Vues

Les vues de notre application sont dans le dossier « views ». Chaque vue de notre application dépend d'une vue qu'on appelle vue générale (V_generale).

C'est dans la vue générale que l'on ajoute le menu, le pied de page ainsi que tous les fichiers js et css nécessaires au fonctionnement de notre application.

Voici la structure de notre V_generale : Dans notre application, chaque vue contient la vue

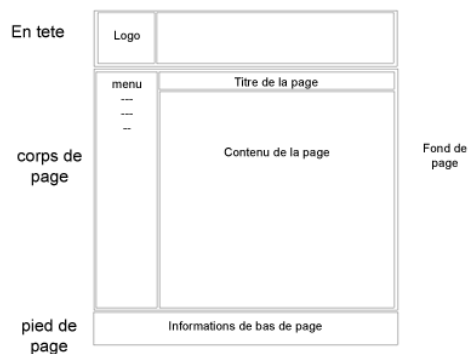


Figure 5 – Vue générale

générale, c'est uniquement la partie contenue de la page qui est modifié.

Voici le code expliquant ce mécanisme : La variable « \$chemin » contient le chemin de la vue

```
<body>
  <?php
    $this->view('/views/V_menu');
    $this->view($chemin);
  ?>
</body>
```

Figure 6 – Code changement de vue

permettant d'afficher les données en fonction des actions de l'utilisateur. C'est pour cela que nous devons définir une variable chemin dans nos différentes méthodes de nos contrôleurs.

H

Rapport de tests

Voici le rapport de tests sur différents fonctionnalités.

Tests	Résultats attendus	Résultats obtenus	Etat de test
Supprimer une activité	Activité supprimé sans erreur de BD	Activité supprimé sans erreur de BD	OK
Supprimer une activité	Les activités dans un parcours qui a un lien de précédence avec elle seront supprimées	Les activités composées dans le parcours qui est précédent ou après cette activité sont aussi supprimées	Ok
Supprimer une activité	Activité a été planifiée, la planification de cette activité sera supprimée	La planification sur cette activité est supprimée	Ok
Supprimer une activité	Onglet correspond à cette activité sera supprimé	Onglet correspond à cette activité est supprimé	Ok
Supprimer un patient	Patient supprimé sans erreur de BD	Patient supprimé sans erreur de BD	OK
Supprimer un patient	Les dossiers parcours affectés à ce patient seront supprimés	Les dossiers parcours affectés à ce patient sont supprimés	Ok

Figure 1 – Tests sur les fonctionnalités

Supprimer un patient	Patient a été planifié, la planification de ce patient sera supprimée	Patient a été planifié, la planification de ce patient est supprimée	Ok
Modifier un patient	Les informations stockées dans la BD seront affichées sur la page	Les informations stockées dans la BD sont affichées sur la page	Ok
Ajouter nouveau RDV	Une page pour prendre le rdv s'affichera	Une page pour prendre le rdv s'affiche	Ok
Modifier un RDV	Une page pour prendre le rdv s'affichera	Une page pour prendre le rdv s'affiche	Ok
Ajouter nouveau RDV	Un nouveau dossier de parcours sera créé dans la BD	Un nouveau dossier de parcours est créé dans la BD	Ok
Ajouter nouveau RDV	Des enregistrements des champs et des onglets associés à ce RDV seront créés dans la BD	Des enregistrements des champs et des onglets associés à ce RDV sont créés dans la BD	Ok
Modifier un RDV	Le dossier de ce rdv sera renouvelé	Le dossier de ce rdv est renouvelé	Ok
Modifier un RDV	Des enregistrements des champs et des onglets associés à ce RDV seront renouvelés	Des enregistrements des champs et des onglets associés à ce RDV sont renouvelés	Ok

Figure 2 – Tests sur les fonctionnalités

Supprimer un RDV	Le dossier sera supprimé	Le dossier est supprimé	Ok
Consulter des dossiers d'un patient	Les dossiers seront affichés par la date de RDV	Les dossiers sont affichés par la date de RDV	Ok
Consulter des dossiers d'un patient	Les onglets présenteront les activités associées à ce parcours	Les onglets présentent les activités associées à ce parcours	Ok
Consulter des dossiers d'un patient	Des champs defaults seront affichés	Des champs defaults sont affichés	Ok
Modifier l'objectif d'un parcours	La modification sera enregistrée	La modification est enregistrée	Ok
Affecter un parcours à patient	Le calendrier pour choisir la date sera coloré par le ratio d'occupation	Le calendrier pour choisir la date est coloré par le ratio d'occupation	Ok

Figure 3 – Tests sur les fonctionnalités

I

Librairies utilisées

Voici la liste des librairies utilisées ainsi que leur version durant ce projet :

- PHP : version 5.5.12
- Bootstrap : version 3.0
- FullCalendar : version 3.0
- MySQL : version 5.6.17



Bibliographie

- [1] Jean COQUELET. « Gestion et Optimisation des Parcours Patients ». Projet Recherche & Développement. Tours, France : Ecole Polytechnique de l'Université François Rabelais de Tours, 2015-2016.
- [2] Guillaume POCHE. « Outil de gestion de parcours patients dans un hôpital de jour ». Projet Recherche & Développement. Tours, France : Ecole Polytechnique de l'Université François Rabelais de Tours, 2016-2017.

Outil de gestion de parcours patient dans un hôpital de jour

Résumé

Dans le cadre de mon projet de recherche et développement, je travaille sur le développement d'une application WEB en collaboration avec l'assistance publique - hôpitaux de Paris. Cet outil doit permettre de visualiser et modifier les plannings du personnel de soins dans un hôpital de jour. Ce document a pour but de définir le contexte et les objectifs et aussi de proposer une analyse permettant de répondre aux besoins du client.

Mots-clés

FullCalendar, planning, Web, Framework PHP, MVC

Abstract

During my final project, I have to develop a tool which is an web application collaborating with APHP (l'Assistance Publique - Hôpitaux de Paris). This application allows patients and staffs in the hospital to see their plan. This document defines the context and objectives of the development and also it propose the analysis which will meet the demande of client

Keywords

FullCalendar, plan, Web, Framework PHP, MVC