

Document d'analyse, estimations et spécifications

Logiciel Healthcare SofTis

19 JUIN 2020

TIS3 -

DA SILVA COSTA Eduarda

VIALLET Camille

BUYSE Anne-Laure



Avant-propos :

Ce dossier permettra d'analyser le système d'information actuel de l'hôpital de Princeton Plainsboro, ainsi que ses spécifications afin d'y apporter des modifications à l'aide de nouvelles fonctionnalités. L'objectif de la conception du logiciel est la maîtrise des coûts et des facturations de Sécurité Sociale pour l'hôpital, ainsi que la gestion de son personnel.

Table des matières

Avant-propos :	2
I. Livable	3
II. Analyse du code fourni	4
1. Diagramme UML	4
2. Commentaire	4
III. Fonctionnalités	5
1. Fonctionnalités fixes	5
2. Fonctionnalités optionnelles :	8

Livrable

Le tableau (figure 1) représente notre estimation en temps et en coût de ce projet. On retrouve les étapes de démarrage telles que la prise en compte du contexte ainsi que le temps prévu à la planification des charges. Au tout début du projet, nous avons choisi de passer une bonne partie de notre temps sur ces tâches, afin de bien comprendre les attentes et de faire les meilleures estimations possibles. Etant donné que nous n'étions que trois étudiantes à travailler sur le projet, ceci joue sûrement un rôle très important lors du calcul des estimations. Dans cette première partie figurent aussi les processus de réalisation des documents, comme le dossier ici présent. Nous estimons tout de même que la majeure partie du temps attribuée à ce projet sera consacrée à l'ajout et au test de nouvelles fonctionnalités, qui ont pour but de rendre le système d'information du CHU Princeton Plainsboro plus intéressant. Enfin, dans les dernières lignes de notre fichier d'estimation, nous retrouvons des valeurs approximatives pour la conception des tests intégratifs et le cahier de recettes, proposant différents scénarios.

En ce qui concerne l'estimation du temps, nous avons utilisé la méthode des trois points. Ce principe évalue la durée passée d'un point de vue optimiste (OPT), pessimiste (PESS) et de la manière la plus probable (PP). Notre but ultime étant de se retrouver entre ces valeurs tout le long du projet.

PROJET : dates début et fin		10-juin	19-juin	TP prévu par période				Charges Prévuees				Coût hor.	Budget
Lot	Livrable	Début	Fin	11-juin	15-juin	18-juin	Opt.	PP	Pess.	Pond.			
1	Démarrage	10-juin	12-juin	26	0	0	20	26	34	26,7			267 €
1.1	Prise en compte contexte et existant	10-juin	12-juin	14	0	0	12,0	14,0	16,0	14,0	10 €		140 €
1.2	CR Réunion démarrage	10-juin	12-juin	3	0	0	2,0	3,0	6,0	3,7	10 €		37 €
1.3	Planification initiale (charges par période)	10-juin	12-juin	9	0	0	6,0	9,0	12,0	9,0	10 €		90 €
2	Réalisation	10-juin	18-juin	9	76	54	88	136	199	141,0			1410 €
2.1	Doc d'Analyse CdC et Spécifications	10-juin	12-juin	9	0	0	6,0	9,0	12,0	9,0	10 €		90 €
2.2	Doc d'architecture	12-juin	18-juin	0	15	11,3	21,0	25,0	33,0	26,3	10 €		263 €
2.3	Notice d'utilisation simplifiée	15-juin	18-juin	0	0	10	9,0	10,0	12,0	10,3	10 €		103 €
2.4	Logiciel et fichiers sources	12-juin	18-juin	0	61	32	52	92	142	95,3			953 €
2.4.1	Fonctionnalités (codage + tests unitaires)	12-juin	18-juin	0	53	32	46	84	132	87,3			873 €
2.4.1.1	Ajout attributs dans Acte	12-juin	14-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.2	Ajout attributs dans Date + gestion du format	12-juin	14-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.3	Affichage listes des patients/médecins	12-juin	14-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.4	Affichage fiches de soins triées	12-juin	14-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.5	Afficher dossierPatient(Patient p)	12-juin	14-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.6	création des classes Personnel, SecrétaireMed et SecrétaireAdmin, MetierCHU et PesonnelHopital	12-juin	15-juin	0	4	0	2,0	4,0	6,0	4,0	10 €		40 €
2.4.1.7	Ajout attributs dans Medecin	12-juin	15-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.8	Ajout attributs dans Patient	12-juin	15-juin	0	2	0	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.9	Interface Accueil	12-juin	17-juin	0	5	0	2,0	4,0	8,0	4,7	10 €		47 €
2.4.1.10	Interface Médecin / Secrétaire Med	12-juin	17-juin	0	15	5	12,0	20,0	32,0	21,3	10 €		213 €
2.4.1.11	Interface Secrétaire Admin	12-juin	17-juin	0	10	10	12,0	20,0	32,0	21,3	10 €		213 €
2.4.1.12	XML	12-juin	17-juin	0	5	5	5,0	10,0	15,0	10,0	10 €		100 €
2.4.1.13	Méthode pour vérifier le n° Sécu	12-juin	17-juin	0	0	4	2,0	4,0	6,0	4,0	10 €		40 €
2.4.1.15	Ajout des tarifs actes fiches de soins (OPTIONNEL)	12-juin	17-juin	0	0	2	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.16	Bouton Impression (OPTIONNEL)	12-juin	17-juin	0	0	2	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.17	Méthode calcul âge + verification format (OPTIONNEL)	12-juin	17-juin	0	0	2	1,0	2,0	3,0	2,0	10 €		20 €
2.4.1.18	Afficher liste médecin selon spécialité (OPTIONNEL)	12-juin	17-juin	0	0	2	1,0	2,0	3,0	2,0	10 €		20 €
2.4.2	Etudes et Outillage gestion sources	12-juin	15-juin	0	8	0	6,0	8,0	10,0	8,0	10 €		80 €
3	Tests et Recette	12-juin	18-juin	0	0	10	11	18	28	19,0			190 €
3.1	Plan de Test	12-juin	15-juin	0	0	3	3,0	6,0	9,0	6,0	10 €		60 €
3.2	Cahier de Recette	12-juin	15-juin	0	0	7	8,0	12,0	19,0	13,0	10 €		130 €
3.3	Tests d'intégration	12-juin	18-juin	0	0	8	3,0	6,0	9,0	6,0	10 €		60 €
4	Management et Réunions Projet	10-juin	19-juin	5	5	12	10,0	15,0	25,0	16,7	10 €		167 €
Totaux				40	81	76	129	195	286	203,3			2033 €

II. Analyse du code fourni

1. Diagramme UML

Ce diagramme représente la rétroconception du logiciel initial du centre hospitalo-universitaire de Princeton-Plainsboro. Il a été réalisé grâce au logiciel Modelio®.

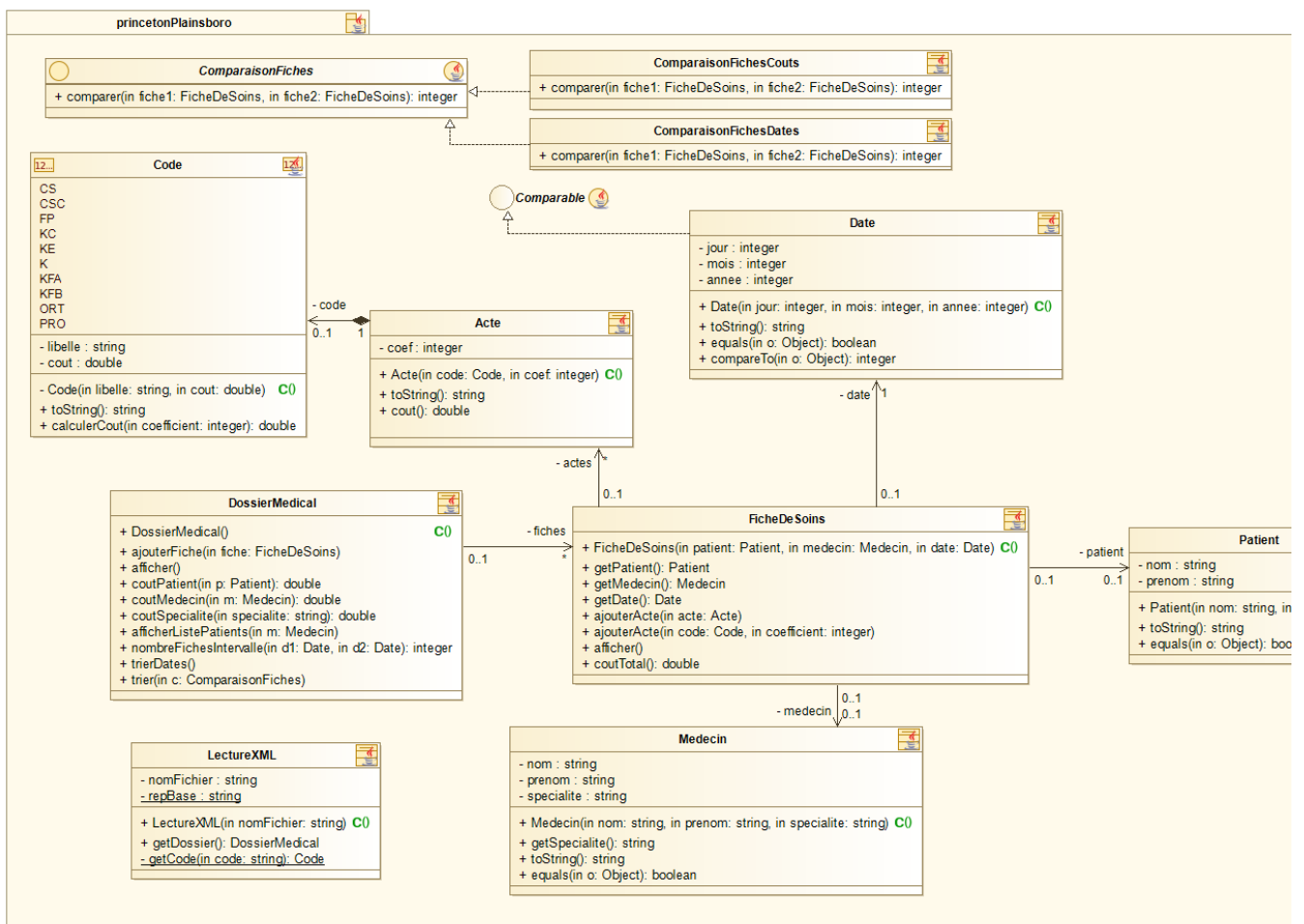


Figure 2: diagramme UML représentant le logiciel initial

2. Commentaire

Ce dossier a pour but d'analyser le système d'information de l'hôpital de Princeton Plainsboro ainsi que ses spécifications, et nous permettra d'expliquer toutes les modifications que nous voulons apporter à l'aide de nouvelles fonctionnalités.

L'hôpital possède déjà un logiciel permettant de gérer les Dossiers Médicaux Informatisés (DMI). Cependant, ce projet a pour objectif de l'améliorer en respectant un cahier des charges actualisé. L'appel d'offre propose de développer davantage l'aspect gestion de coût et de facturation de l'hôpital à la Sécurité Sociale du logiciel.

Afin d'étudier le logiciel initial, nous avons effectué une rétroconception du code existant. Le diagramme UML précédent permet de le clarifier (*figure 2*).

Initialement, lorsque le programme est lancé, nous avons accès à un unique dossier médical qui propose une liste de fiche de soins. Cette liste permet d'identifier le patient concerné, le médecin associé, la date de réalisation et l'ensemble des actes médicaux qui ont été réalisés. L'acte médical est défini par son libellé et son coût. Le médecin est caractérisé par son nom et sa spécialité. Les informations relatives au patient sont son nom et son prénom.

Ce logiciel permet à l'utilisateur de réaliser les actions suivantes :

- Afficher la liste des patients d'un médecin donné
- Afficher les informations liées au patient et médecin (énumérées précédemment)
- Créer de nouvelles fiches de soins
- Calculer le coût d'un acte
- Calculer les coûts totaux pour un patient
- Calculer les coûts totaux par médecin
- Calculer les coûts totaux par spécialité

III. Fonctionnalités

D'après les informations données dans le cahier de charges, nous avons choisi de relever les fonctionnalités suivantes, afin de créer un système d'information plus adéquat aux demandes. Nous avons basé nos choix sur le fait que nous ne sommes que trois étudiantes à travailler sur le projet, mais que nous voulions fournir à notre client la réponse la plus adéquate à ses demandes.

1. Fonctionnalités fixes

Pour les fonctionnalités que nous appellerons ‘fixes’, nous nous engageons à les développer en respectant les souhaits de nos clients.

- Ajout d’attributs dans la classe Acte :
 - String nomActe : renseigne le nom de l’acte.
 - String nomMed : renseigne le nom du médecin impliqué dans l’acte.
 - Date date : donne la date sous le format ‘jour-mois-année à heure : mins’
 - Type type : énumération du type d’acte médical (diagnostique ou thérapeutique).
 - String comm : éventuels commentaires du médecin sur l’acte médical.
- Vérifier tous les prix des actes médicaux et en rajouter d’autres dans la classe Code.
- Ajout d’attributs dans la classe Date et gestion du format :
 - int heures : donne l’heure pour la date demandée.
 - int minutes : donne les minutes pour la date demandée.
 - verifierDate() : vérifier si la date inscrite respecte le format demandé.
- Affichage de la liste des patients/médecins :
 - afficherListePatients() : affiche la liste des patients d’un certain médecin.
 - getListeMedecins() : affiche la liste des médecins présents dans la classe DossieMedical.
- Affichage des fiches de soins triées :
 - Coût décroissant, numéro de fiche de soins décroissant et date décroissante → utilisera la méthode trierDecroissant(ComparaisonFiches c, List<FicheDeSoins> feuilleSoins) : affiche la liste des fiches de soins triées selon leur coût, date ou numéro de fiche de soins (ordre décroissant).
 - Entre deux dates → fichesTriees2Dates(Date dateDebut, Date dateFin): affiche la liste des fiches de soins effectuées entre deux dates données.
- Affichage du Dossier Patient : getFichesPatient(Patient p) : affiche le dossier patient contenant ses fiches patient.
- Création des classes Personnel, SecretaireMed, Secretaire Admin, MetierCHU, et PersonnelHopital : ces classes nous permettront de créer l’accès au système d’information à différentes personnes composant le personnel de l’hôpital (médecin, secrétaire médicale et secrétaire administrative), et ainsi de modéliser tout le personnel y travaillant. De plus, la classe Personnel aura des attributs que tout le personnel de l’hôpital héritera (Medecin, SecretaireMed, SecretaireAdmin) : le prénom, le nom, l’identifiant, le mot de passe et le numéro de téléphone.
- Ajout d’attributs dans la classe Médecin :
 - String telephone : donne le numéro de téléphone du médecin.

-
- String mdp : mot-de-passe associé au médecin.
 - Specialite Specialite : la spécialité du médecin ne sera plus un String mais une énumération.
 - Une particularité pour l'identifiant du médecin : ceci correspondra à son **numéro RPPS**, attribué par le Répertoire Partagé des Professionnels de Santé, étant unique et conservable pendant toute leur carrière.
-
- Ajout d'attributs dans la classe Patient :
 - String adresse : renseigne l'adresse du patient.
 - Date dateDeNaissance : renseigne la date de naissance du patient, sous le format JJ-MM-AAAA HH:MM. Si l'heure et les minutes sont inconnues, la date sera affichée sans ces données (méthode toStringDate() dans la classe Date).

Attention, un message d'erreur s'affichera si la date est incorrecte.
 - String numeroSecuriteSociale : renseigne le numéro de sécurité sociale.
 - double poids : renseigne le poids du patient.
 - double taille : renseigne la taille du patient.
 - Vérifier le numéro de la sécurité sociale : boolean verifierSecu() : cette méthode permet de vérifier que le numéro de sécurité sociale inscrit est valable et conforme aux règles données dans le cahier des charges. Un message d'erreur s'affichera si cela n'est pas respecté.
 - Création de classes Java capables de gérer/lire le document XML existant (dossiers.xml), et les autres documents XML (personnels.xml) que nous créerons pour faire fonctionner correctement notre logiciel.
 - Ceci nous permettra une bonne gestion du personnel et des dossiers médicaux.
 - Interface Accueil (1) : cette interface représente la page de connexion du personnel du CHU Princeton-Plainsboro. La personne devra se connecter avec son identifiant et son mot de passe (tous les deux uniques).
 - Interface Médecin/Secrétaire médicale (2.a) : cette interface est seulement disponible pour l'équipe médicale. Ils peuvent voir leur identification, la liste des patients ainsi que leurs dossier médicaux, ainsi que des boutons pour ajouter des actes et des nouveaux patients.
 - Interface Secrétaire Administrative (2.b) : cette interface est disponible du côté administratif du personnel. La liste de patients peut être consultée, ainsi que le coût par spécialisation, par patient, par acte médical, par fiche de soins et par médecin. Il y a la possibilité de cliquer sur une fiche de soins pour en dévoiler plus, ainsi que trier ces listes.

-
- Interface Dossier Patient (3.a.1) : à partir de cette interface, la personne peut saisir de nouveaux actes médicaux par rapport à un patient précis, ainsi que de nouvelles fiches de soins.
 - Interface Formulaire Inscription (3.a.2) : à partir de cette interface, un nouveau patient peut être ajouté à la liste des patients d'un médecin et à la liste de tous les patients de l'hôpital.
 - Interface Facture (3.b) : dans cette interface, la secrétaire trouvera la liste des actes correspondant au patient, les médecins réalisant les actes et leurs prix.

Fonctionnalités optionnelles :

Nous nous engageons à développer ces fonctionnalités si le temps nous le permet.

- Dans l'interface administrative, rajouter un onglet qui permet la gestion du personnel (donc une liste complète de ce dernier), avec la possibilité de créer une nouvelle personne travaillant à l'hôpital Princeton Plainsboro.
- Vérification des années bissextiles pour la date : `verifierBisDate()` : vérifier la cohérence entre l'année indiquée et le reste de la date.
- Ajout de l'impression : bouton 'Imprimer' : ajouter le bouton à l'interface 3.a.1, afin d'imprimer le dossier d'un patient.
- Calcul âge + vérification de son format : `calculAge()` : calculer l'âge du patient à partir de sa date de naissance.
- Afficher la liste des médecins selon leur spécialité : afficher la liste de tous les médecins correspondant aux spécialités.
- Ajouter plus de spécialités à l'hôpital.

