## RESUME

L’objectif de ce projet est de mettre à la disposition du centre médical interentreprises(CMI) de Orange cote d’Ivoire Telecom(OCIT) une nouvelle application de gestion médicale. Cette nouvelle application prend en compte les besoins du centre qu’ils soient de la gestion des activités du circuit curatif, des activités du circuit préventif, de la gestion du stock pharmaceutique et la gestion des statistiques.

L’étude globale de la solution existante nous a permis de proposer des solutions appropriées. Nous avons procédé dans un premier temps à la formalisation des besoins du CMI qui s’est aboutie à la modélisation de ceux-ci. Ensuite nous avons procédés à la réalisation du projet en s’inspirant des modèles. Enfin des tests ont été réalisés sur la nouvelle solution avant la mise en production et la formation des Agents du CMI.

Soulignons que cette application permettra à Orange côte Télécom de suivre en temps Réel l’état de santé de ses agents, et quant au CMI elle lui permettra de gérer aisément le traitement d’un patient.

# 

# ETUDE CONCEPTUELLE

ents modèles qui décrieront notre c

hode d’analyse et de c

## CHAPITRE III- METHODE D’ANALYSE ET DE CONCEPTION

Une méthode d’analyse est un ensemble de démarches que suit l'esprit et l'arrangement qui en résulte*.* C’est donc un ensemble formé de modèles ou de langage et d’une démarche ou processus qui utilise ces modèles. En clair il s’agit de représenter le domaine à étudier ou à automatiser à l’aide de langage abstrait, en suivant des étapes bien précises, avec chaque étapes un modèles bien défini à élaborer. Son objectif étant de formaliser les étapes préliminaires de la conception d’un système informatique afin de rendre son développement plus fidèles aux besoins du client. Il existe plusieurs méthodes d’analyse et de conception : les plus utilisées sont la méthode MERISE et le processus Unifié que nous tenterons de décrire dans la suite.

### III.1- Présentation des méthodes d’analyse et de conception

#### III.1.1- Présentation du processus unifié(UML)

Le processus unifié est une méthode d’analyse et de conception générique de développement de logiciel. C’est une méthode :

* Pilotée par les cas d’utilisation d’UML
* Elle s’appuie sur un cycle de vie itératif et incrémental, Chaque itération porte sur un niveau d’abstraction de plus en plus précis.
* Centrée sur l’architecture
* Orientée vers la diminution des risques
* Il est construit autour de la création et de la maintenance d’un modèle, plutôt que de la production de montage de documents.
* Il est orienté composant.
* Il est orienté utilisateur.

De plus le processus unifié utilise une démarche en quatre (4) phases

* Démarche(Lancement) : phase d’initialisation du projet où l’on mène une étude d’opportunité et de faisabilité du système à construire, consiste aussi à l’indentification des cas d’utilisation.
* Élaboration : spécification du plan du projet, des exigences et des bases de l’architecture. Le plan est spécifié en un plan d’itérations.
* Construction : réalisation du produit
* Transition. Test Validation et déploiement.

Il utilise comme langage de modélisation **l’UML** (Unified Modeling Language) qui propose 13 diagrammes dans sa version 2.0 consigné dans le tableau suivant.

|  |  |  |
| --- | --- | --- |
| Diagrammes structurels(6) | Diagrammes fonctionnels et comportementaux(7) | |
| * Digramme de classe * Digramme d’objet * Diagramme de composant * Digramme de déploiement * Digramme de paquetage (UML 2) * Digramme de structure composite UML 2 | | * Digramme de cas d’utilisation * Digramme de séquences * Digramme d’activités * Digramme de communication * Digramme d’état transition * Digramme de temps UML 2 * Diagramme global d’interaction UML 2 |

Tableau 2: Diagrammes UML 2

Le processus unifié étant générique, il faut l’adapter au projet et à l’environnement de travail. En ce sens il existe plusieurs implémentations dont les principaux sont: RUP, XUP et 2TUP. Nous les présenterons dans le tableau récapitulatif suivant.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Description | Points fort | Point faible |
| Cascade | -Propose de dérouler les phases projet de  Manière séquentielle  - Cité pour des  raisons historiques | - Distingue  clairement les phases projet | - Non itératif  - Ne propose pas de  modèles de documents |
| RUP : Rational Unified Process | - Promu par  Rational.  - Le RUP est à la  fois une méthodologie et un outil prêt à l'emploi (documents types partagés dans  un référentiel Web)  - Cible des projets de  plus de 10 personnes | - Itératif  - Spécifie le dialogue  entre les différents  intervenants du  projet : les livrables,  les plannings, les  prototypes…  - Propose des modèles de documents, et des canevas pour des projets types  - Propose des modèles de documents, et des  canevas pour des  projets types | - Coûteux à  personnaliser  - Très axé processus, au détriment du  développement : peu de place pour le code et la technologie |
| XP eXtreme  Programming | - Ensemble de «  Bests Practices » de  développement  (travail en équipes, transfert de  compétences…)  - Cible des projets de  moins de 10  personnes | - Itératif  - Simple à mettre en  œuvre  - Fait une large place  aux aspects  techniques :  prototypes, règles de  développement, tests… | - Ne couvre pas les  phases en amont et en  aval au développement :  capture des besoins,  support, maintenance,  tests d'intégration…  - Elude la phase  d'analyse, si bien qu'on  peut dépenser son  énergie à faire et défaire  - Assez flou dans sa  mise en œuvre: quels  intervenants, quels  livrables ? |
| 2TUP :Two Track  Unified Process | - S'articule autour de  l'architecture  - Propose un cycle de  développement en Y  - Détaillé dans "UML  en action" (voir  références)  - Cible des projets de  toutes tailles | - Itératif  - Fait une large place  à la technologie et à la  gestion du risque  - Définit les profils  des intervenants, les  livrables, les  plannings, les prototypes | Plutôt superficiel sur  les phases situées en  amont et en aval du  développement : capture des besoins, support,  maintenance, gestion du changement…  - Ne propose pas de  documents types |

Tableau 3: Description des implémentations du PU

#### III.1.2- Présentation de la méthode Merise

La méthode MERISE (Méthode d’Etude et de Réalisation Informatique pour les Systèmes d’Entreprise), créée dans les années 70, a pour objectif de fournir à la fois une démarche, des modèles, des formalismes et des normes pour la conception et la mise en place du système d’information d’une entreprise. La méthode MERISE est basée sur la séparation des données et des traitements à effectuer en plusieurs modèles conceptuels et physiques. Cette séparation des données et des traitements assure une longévité au modèle. En effet, l'agencement des données n'a pas à être souvent remanié, tandis que les traitements le sont plus fréquemment. La puissance de la méthode réside dans le fait qu’elle permet de schématiser les niveaux d’abstraction et offre un niveau de granularité adaptable à tous les besoins.

La méthode MERISE utilise :

* **un modèle fonctionnel** basé sur les diagrammes de flux (flux d’informations ou messages) ;
* **un modèle statique** basé sur l’Entité-Association enrichi de méthodes de traitement. Il décrit les tâches à effectuer à la réception ou pour l’émission d’un flux d’informations;
* **un modèle dynamique** des objets explicitant le contrôle et les interactions des objets. C’est la structure de mémorisation des informations représentée sous une forme qui permet un passage aisé vers les enregistrements informatiques.

##### III.1.2.1- Les niveaux

L’informatique consiste à mettre à disposition de l’utilisateur des moyens ou des outils de gestion informatique. Avant de spécifier les moyens informatiques, il est nécessaire de définir le travail de ce ou de ces utilisateurs finaux, de définir l’organisation, l’analyse des objectifs et des fonctions majeures de l’entreprise doit être menée. Ainsi, l’informatisation est conçue en fonction de l’organisation et l’organisation en fonction des objectifs à atteindre. Or l’enchainement de l’informatique, de l’organisation et de la fonction nécessite un découpage en niveaux de la démarche d’informatisation.

Ces niveaux sont :

* **Le niveau conceptuel**

C’est le niveau le plus invariant, il correspond aux finalités de l’entreprise. Il s’agit de décrire le « Quoi » en faisant abstraction des contraintes d’organisation et techniques. Les modèles utilisés pour la description conceptuelle du système d’information sont :

* le **Modèle Conceptuel de Communication (MCC)** qui représente les échanges de flux de produits, d’énergie, de personne, de valeur ou d’information entre système.
* le **Modèle Conceptuel de Données (MCD)** qui est une description des données et des relations et est réalisé à l’aide des trois concepts du formalisme entité-association : entité (ou objet) – relation – propriétés.
* le **Modèle Conceptuel des Traitements (MCT)** qui est la description de la partie dynamique du système d’information et est réalisé à l’aide des concepts suivants : événements/résultat – synchronisation – opération - processus.
* **Le niveau organisationnel**

Les choix d’organisation sont pris en compte à ce niveau comme la répartition des traitements entre l’homme et la machine, le mode de fonctionnement (temps réel ou différé) et l’affectation des données et des traitements. Ce niveau répond aux questions « Qui fait Quoi et Où ».

Trois modèles sont associés à ce niveau :

* le **Modèle Organisationnel de Communication (MOC)** qui représente les échanges qui ont lieu entre les sites de traitements et de données. Il ne concerne que les communications entre sites. Il n’existe pas s’il n’existe qu’un site ;
* le **Modèle Organisationnel de Données (MOD)** qui reprend le formalisme utilisé dans le MCD et qui représente aussi l’ensemble des données par type de site organisationnel ;
* le **Modèle Organisationnel des Traitements (MOT)** qui représente par procédures les phases et les tâches exécutées par chaque poste de travail.
* **Le niveau logique**

Ce niveau décrit le système d’information au plan informatique sans choix de matériel ou de logiciel précis. Nous avons trois modèles reliés à ce niveau :

* le **Modèle Logique de Communication (MLC)** qui est une représentation des messages échangés entre site et base de données. Il provient du MLD et de l’utilisation des outils en temps différé.
* le **Modèle Logique des Données (MLD)** qui est une représentation des données décrit dans le MCD en tenant compte du type de base de données.
* le **Modèle Logique des Traitements (MLT)** qui permet de décrire la conception technique qui traite principalement de la structuration en unités de traitement de type temps réel ou de type temps différé.
* **Le niveau physique**

Ce niveau décrit le résultat de la méthode ou l’informatisation finale. Il dépend des logiciels de développement nécessaires à la programmation et à la manipulation des données. La méthode laisse place aux normes du réel.

Nous avons encore trois modèles liés :

* **le Modèle Physique des Communication (MPC)** qui comprend la télématique entre sites informatiques ;
* **le Modèle Physique des Données (MPD)** qui n’est qu’un modèle de la base de données;
* **le Modèle Physique des Traitements (MPT)** qui comprend les programmes techniques et leur environnement d’exploitation, moniteur temps réel, traitement par lot, temps partagé.

##### I.1.2.2- La démarche

La méthode MERISE propose une démarche de construction de système d’information en 6 étapes:

* **le schéma directeur** : définition de la politique de l’entreprise ;
* **l’étude préalable** : analyse de l’existant et étude des différentes solutions possible puis choix de la solution appropriée ;
* **l’étude détaillée** : étude approfondie de la solution choisie par la direction ;
* **l’étude technique** : spécifications techniques complètes, seuls les informaticiens interviennent ;
* **la réalisation** : écriture des programmes, tests, essais, formation utilisateur ;
* **la maintenance** : suivi et évolution.

Cette démarche s’appuie sur un cycle de vie en cascade qui date de 1970 et est l’œuvre de Royce. Il s’agit d’un mode linéaire où toute étape est censée n’avoir de rétroaction que sur l’étape qui la précède.

L’activité d’une étape se réalise avec les résultats fournis par l’étape précédente ; ainsi, chaque étape sert de contrôle du travail effectué lors de l’étape précédente. L’utilisateur attend le déroulement complet du cycle de vie du logiciel pour vérifier, lors de la dernière étape, l’adéquation entre ses exigences et le produit livré.

Schéma Directeur

Etude Détaillé

Etude Préalable

Etude Technique

Etude Réalisation

Maintenance

**Cycle de vie**

Figure 2: la démarche Merise

### III.2- Comparaison processus unifié et Merise

|  |  |  |
| --- | --- | --- |
| CRITERES | MERISE | PROCESSUS UNIFIE |
| Données et traitements | Séparation des données et des traitements. | Regroupement des données et méthodes au sein des classes. Application du principe de l’encapsulation. |
| Niveaux | Plusieurs niveaux : conceptuel, logique, organisationnel, physique avec plusieurs types de modèles : données, traitements, communication. Existence de règles de passage entre les différents niveaux. | Niveau unique mais plusieurs types de modèles en fonction de l’aspect à décrire. Affinement des modèles lors des différentes étapes de l’analyse et de la conception. Continuité entre les différentes phases d'élaboration de l'application (traçabilité). |
| Gestion de projet | Concentré principalement sur la conception, sans prise en compte de la phase de programmation (modèle de déploiement et de composant absent) | Prise en compte de tous les stades de la conduite d'un projet.  Cohérence de la phase d’analyse jusqu’au code du programme. |

Tableau 4: Etude comparative entre la méthode MERISE et Processus unifié

### III.3- Choix de la méthode

L’analyse du tableau ci-dessus permet de faire ressortir plusieurs points. Mais l’élément majeur est que, MERISE offre une démarche d’analyse cohérente et rigoureuse, et mieux orientée vers les BD relationnelles. Elle est généralement utilisée dans l’informatique de gestion. Alors que le processus unifié utilisant UML qui est basé sur l’approche objet, est très souvent utilisé dans l’informatique technique (temps réel) et dans les projets évolutifs c’est à dire pouvant avoir différentes versions, formalise les interactions entre le système et l’utilisateur. Vu les besoin exprimé dans le cahier de charge, celui de favoriser l’ergonomie, la simplicité d’utilisation, la livraison de l’application en plusieurs version et vu l’interaction utilisateur/système, nous choisissons le processus unifié comme méthode de conception après avoir modélisé notre problème avec le langage UML.

* **Choix de la variante du PU**

Pour ce projet nous avons choisirons le Two Track Unified Process (2TUP) car il offre les possibilités suivantes :

* Itératif ;
* Offre des privilèges aux interactions utilisateurs ;
* Prise en compte les contraintes liées aux changements des SI
* Possibilité de décomposer le problème suivant un axe fonctionnel et un axe technique.
* Adapter à la représentions systèmes sous forme de module ;

Voici décrit schématiquement le processus 2TUP.



Figure 3 : Le processus 2TUP

Dans la suite nous nous servirons de cette méthode pour faire la modélisation métier de notre système.