**DOCUMENT DE SPECIFICATION DU SYSTEM**

|  |
| --- |
| Par |
| NDANG ESSI PIERRE JUNIOR |
| ESSOUMA MBARGA VALERIE CONSTANCE |
| KOUABITCHOU NJIECHEU RAPHAËL ANGE |

**SOMMAIRE**

1. Introduction
   1. L’objectif du document des exigences
   2. Le porté du produit
   3. Définition Acronyme et abréviation
   4. Référence
   5. Aperçue du reste du document
2. Description générale
   1. Perspective du produit
   2. Fonctions du produit
   3. Caractéristique des utilisateurs
   4. Contrainte générale
   5. Contraintes de design et d’implémentation
   6. Hypothèse et dépendance
3. Exigences spécifiques
   1. Exigences fonctionnelles
4. Exigences non fonctionnelles
   1. Exigences lies à la performance
   2. Exigences lies à la sécurité
   3. Attributs de qualité du système
5. Annexe
6. Index

**Prérequis :**

System’s model document + System’s and users’ specifications document

1. **Introduction**
   1. **L’objectif du document des exigences**

Ce document a pour but ; d’expliquer comment le système va se comporter a de manière non technique (une personne ne faisant pas l’informatique puisse comprendre) et de faciliter sa conception.

* 1. **Le porté du produit**

Ce logiciel devra être utilise dans n’importe quelle entreprise professionnel (où la gestion est faite par une personne formée), il ne sert à rien dans un autre cadre que celui-ci.

* 1. **Définition Acronyme et abréviation**

Aucun

* 1. **Référence**
* INF2013 – Chapitre 3. Exigences Systèmes et utilisateur.
* Ques10.com
  1. **Aperçue du reste du document**

1. **Description générale**
   1. **Perspective du produit**

Faciliter la gestion des salaires dans une entreprise professionnelle.

* 1. **Fonctions du produit**
     1. Le système doit pouvoir enregistrer un employé, et stocker ses informations dans une base de données accessible facilement. (Une base de données intégré au système n’impose pas d’utiliser un logiciel externe)
     2. Le système doit pouvoir calculer le nombre d’années de travail d’un employé. (Puisque celui-ci peut changer, selon les critères de calcul, on ne souhaite pas le conserver dans la base de données)
     3. Le système doit pouvoir calculer le salaire d’un employé. (Puisque celui-ci peut changer, selon les critères de calcul, on ne souhaite pas le conserver dans la base de données)
     4. Le système doit déterminer le statut d’un employé dans l’entreprise
     5. Le système doit pouvoir mettre les informations d’un employé a jour et effectuer la mise a jour dans la base de données simultanément
     6. Le système doit pouvoir supprimer un employé de la base de données (on ne souhaite pas garder les informations des employés licenciés par exemple).
     7. Le système doit pouvoir fournir les informations sur une entreprise. (Pour faire un sondage ou une statistique quelconque par exemple)
  2. **Caractéristique des utilisateurs**

Les utilisateurs de ce produit sont des personnes responsables de la gestion des salaires. Ici on a **La direction des resources humaines ET le ministère de la fonction publique.**

* 1. **Contrainte générale**
     1. **Exigences de faisabilité**
        1. Peut-on faire ce logiciel ?

OUI, il a un cout très léger.

* + - 1. A-t-on les fonds nécessaire ?

Calculons les fonds en FCFA en utilisant la **méthode par analogie**. On va donc comparer le cout en se basant sur un projet similaire qu’on a fait en utilisant le langage C. Il était question de gérer les étudiants d’un concours et permettre la réalisation d’un ensemble de fonctionnalités parmi lesquels ; admis, ajourne, ajouter un étudiant … Le cout de ce projet fut de 0FCFA. On conclut que celui-ci ne nous demandera pas de fonds.

* + - 1. Est-ce la bonne période ?

Etant donné que c’est un projet d’apprentissage, on n’a pas à vérifier s’il est actuellement demandé sur le marché.

* + - 1. Qu’est ce qu’on gagne en le faisant ?

Puisque c’est un projet d’apprentissage, on gagne en expérience pratique.

* + 1. **Fonctions d’audit**

Aucun copyright, logiciel open source.

* 1. **Hypothèse et dépendance**

1. **Exigences spécifiques**
   1. **Exigences fonctionnelles**

|  |
| --- |
| Exigence utilisateur |
| 1. Le système doit pouvoir enregistrer un employé, et stocker ses informations dans une base de données accessible facilement. |
| Exigence système |
| 1. Lorsqu’on souhaite enregistrer un employé, le système doit envoyer un formulaire qui enregistre : son nom, son prénom, son matricule, sa date de naissance, son lieu de naissance, sa nationalité, son statut matrimonial, sa fonction, son type de contrat, sa date de recrutement, sa date de fin d’emplois, son salaire de base, son salaire net. 2. Le système doit stocker les informations recueillies dans la table employé de la base de données du système. |
| Exigence utilisateur |
| 1. Le système doit pouvoir calculer le nombre d’années de travail d’un employé |
| Exigence système |
| 1. Lorsqu’on souhaite calculer le nombre d’années d’un employé, le système doit permettre à l’utilisateur d’entrer le matricule de l’employé en question. 2. Le système doit effectuer une recherche de cet employé dans la base de données. 3. Le système fait ce calcul en utilisant les dates de recrutement et de fin d’emplois présent dans la table employée, dans la base de données. 4. En complément a ce calcul, le système doit pouvoir fournir ces statistiques ; les primes et les gratifications, les heures supplémentaires, les retenues pour absence. 5. Le système doit afficher les résultats à l’utilisateur |
| Exigence utilisateur |
| 1. Le système doit pouvoir calculer le salaire d’un employé |
| Exigence système |
| 1. Lorsqu’on souhaite calculer le salaire d’un employé, le système doit permettre à l’utilisateur d’entrer le matricule de l’employé en question. 2. Le système doit effectuer une recherche de cet employé dans la base de données. 3. Le système fait ce calcul en utilisant les TVA et l’indemnité compensatoire des congés payés. 4. Le système doit afficher le résultat de la requête a l’utilisateur |
| Exigence utilisateur |
| 1. Le système doit déterminer le statut d’un employé dans l’entreprise |
| Exigence système |
| 1. Lorsqu’on souhaite connaitre le statut d’un employé, le système doit permettre à l’utilisateur d’entrer le matricule de l’employé en question. 2. Le système doit effectuer une recherche de cet employé dans la base de données. 3. Le système doit ressortir le statut matrimonial de l’employé. 4. Le système doit afficher le résultat de la requête a l’utilisateur |
| Exigence utilisateur |
| 1. Le système doit pouvoir mettre les informations d’un employé à jour |
| Exigence système |
| 1. Lorsqu’on souhaite mettre un employé a jour, le système doit permettre à l’utilisateur d’entrer le matricule de l’employé en question. 2. Le système doit effectuer une recherche de cet employé dans la base de données. 3. Le système permet à l’utilisateur d’entrer les informations à mettre à jour. 4. Le système doit également faire la mise à jour dans la base de données. |
| Exigence utilisateur |
| 1. Le système doit pouvoir supprimer un employé de la base de données |
| Exigence système |
| 1. Lorsqu’on souhaite supprimer un employé, le système doit permettre à l’utilisateur d’entrer le matricule de l’employé en question. 2. Le système doit effectuer une recherche de cet employé dans la base de données. 3. Le système doit enlever toutes les informations de cet employé de la base de données. 4. Le système doit également faire la mise à jour dans la base de données. |
| Exigence utilisateur |
| 1. Le système doit permettre d’enregistrer une entreprise. |
| Exigence système |
| 1. Lorsqu’on souhaite enregistrer une entreprise, le système doit permettre à l’utilisateur d’entrer ; le nom, l’année de création, le siège social, la boite postale, le(s) téléphone(s), le fax, l’Email, l’activité, l’effectif, le capital, le site internet, le nom du directeur, le numéro contribuable, la forme juridique, le chiffre d’affaire, de l’entreprise en question. 2. Le système doit stocker les informations recueillies dans la table entreprise de la base de données du système. |

1. **Exigences non fonctionnelles**
   1. **Exigences lies à la performance**

System exploitation: Windows, Linux

Bits: 32 ou 64

RAM: 2GO minimum

* 1. **Exigences lies a la sécurité**

Aucune

* 1. **Attributs de qualité du system**

1. **Annexe**
   1. **Durée du projet**

**Pour déterminer la durée du projet, nous allons utiliser la méthode PERT.**

En guise de rappel, le PERT se fait en 7 étapes, à chaque étape, on produira la solution.

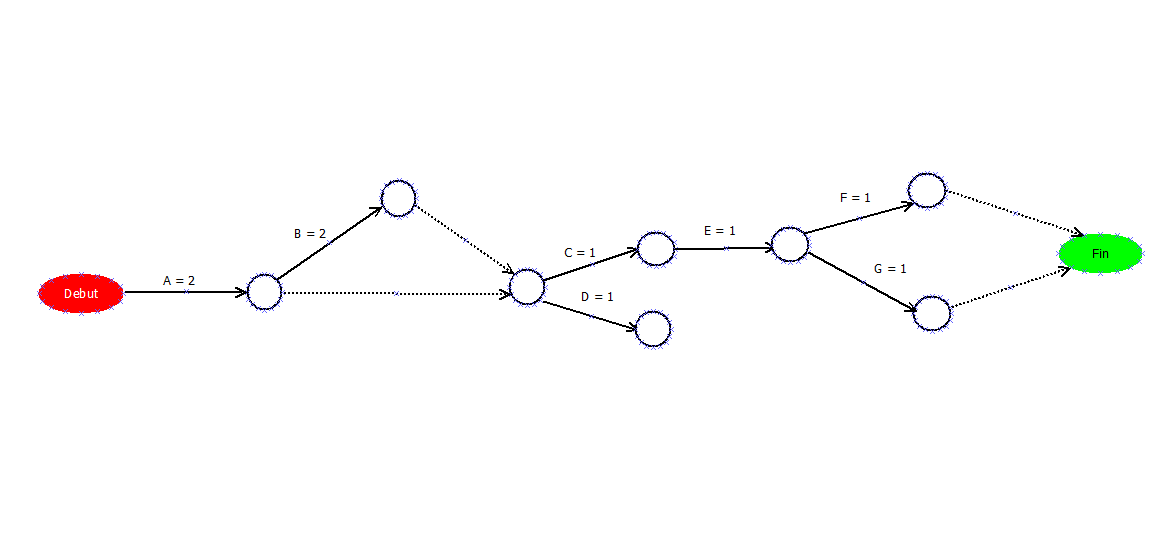
1. Déterminer la liste des taches nécessaires a la réalisation du projet et estimer leur durée.

|  |  |
| --- | --- |
| Tache | Durée estimée |
| Renseignement des informations sur l’entreprise (A) | 2 jours |
| Enregistrement d’un employé (B) | 2 jours |
| Calcul du nombre d’année de travail d’un employé (C) | 1 jour |
| Calcul des salaires (D) | 1 jour |
| Statut de l’employé (E) | 1 jour |
| Mise à jour d’un employé (F) | 1 jour |
| Suppression d’un employé (G) | 1 jour |

1. Déterminer les dépendances logiques entre les taches

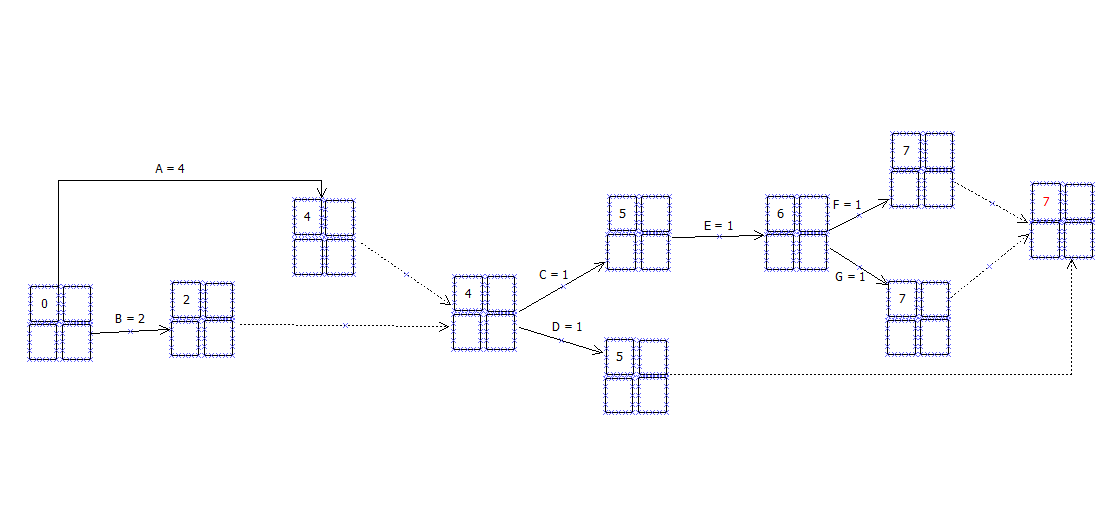
|  |  |  |
| --- | --- | --- |
| Tache | Durée estimée (jours) | Prédécesseur |
| A | 4 | - |
| B | 2 | - |
| C | 1 | A, B |
| D | 1 | A, B |
| E | 1 | C |
| F | 1 | E |
| G | 1 | E |

1. Dessiner le PERT

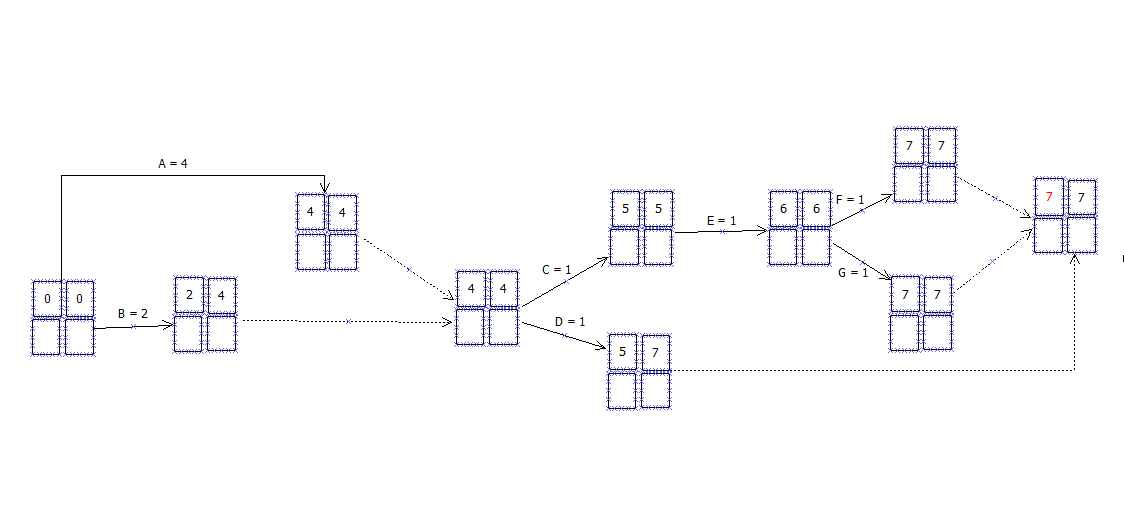


1. Calculer pour chaque nœud la date au plus tôt

D’après le diagramme suivant, le projet va se terminer au plut tôt en 7 jours.



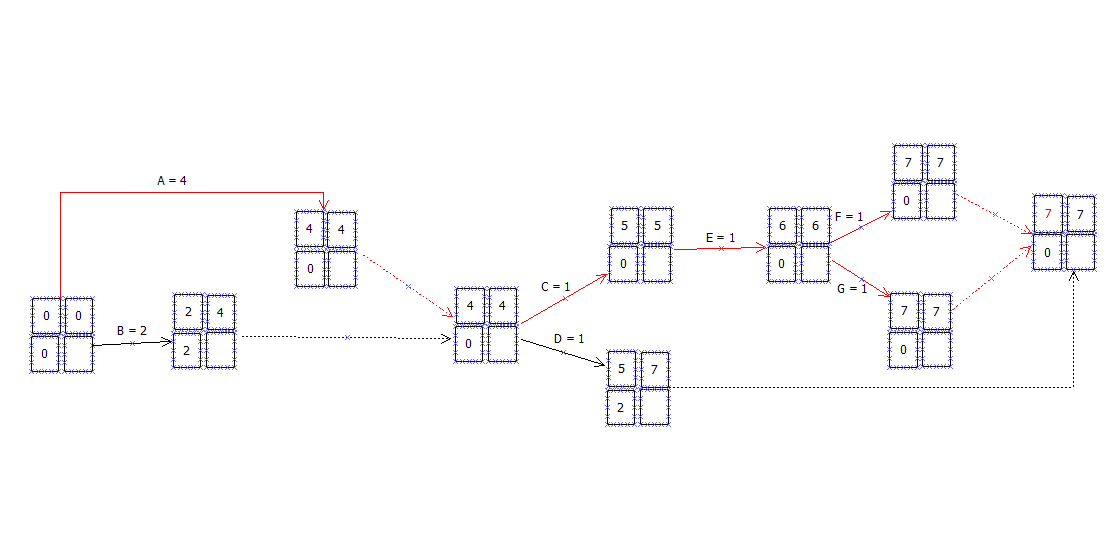
1. Calculer pour chaque nœud la date au plus tard



1. Déterminer le chemin critique

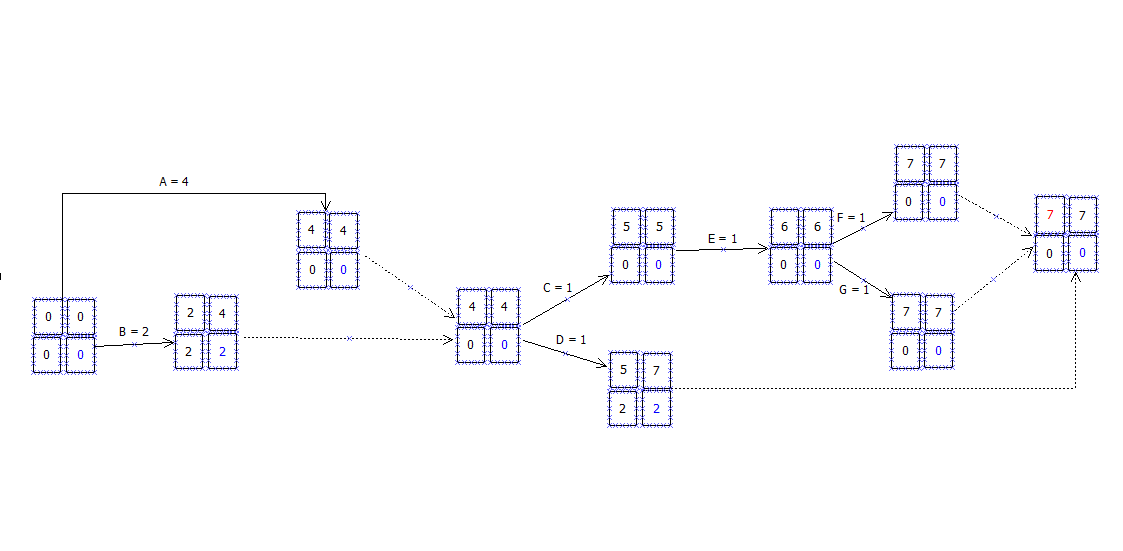
Ce chemin est le chemin qui dans lequel aucun retard ne sera toléré. Il peut en exister plusieurs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tache | Date de début au plus tôt | Durée estimée (jours) | Date de fin au plus tard | Marge totale |
| A | 0 | 4 | 4 | 0 |
| B | 0 | 2 | 4 | 2 |
| C | 4 | 1 | 5 | 0 |
| D | 4 | 1 | 7 | 3 |
| E | 5 | 1 | 6 | 0 |
| F | 6 | 1 | 7 | 0 |
| G | 6 | 1 | 7 | 0 |

  
On remarque qu’on a 2 chemins critiques.

1. A, C, E, F
2. A, C, E, G
3. Calculer les marges libres

Ici, on calcule le délai de retard qu’une tache peut prendre sans provoquer de retard a aucun de ses successeurs.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tache | Date de début au plus tôt | Durée estimée (jours) | Date de fin au plus tard | Marge totale | Marge libre |
| A | 0 | 4 | 4 | 0 | 0 |
| B | 0 | 2 | 4 | 2 | 2 |
| C | 4 | 1 | 5 | 0 | 0 |
| D | 4 | 1 | 7 | 3 | 2 |
| E | 5 | 1 | 6 | 0 | 0 |
| F | 6 | 1 | 7 | 0 | 0 |
| G | 6 | 1 | 7 | 0 | 0 |

1. **Index**