Cahier de charge d’une application CheckUp

1. Présentation du projet :

En appliquant nos connaissances de base en GL,java,framework PHP :synfony4 et base de données et dans le but de réalisation d’un projet au niveau du module projet intégré développement web java, nous envisageons de réaliser une application web et desktop de checkup ayant 2 clients web et java qui permet la gestion d’une application médicale dédiée au cadre médical et aux différents patients en assurant toutes les fonctionnalités de base lui permettant de bien gérer ses différents services.

1. Objectif du projet :

L’application checkup que nous visons à réaliser offre à la clientèle de l’application web un ensemble de services. Pour satisfaire mieux les exigences de chaque client, l’application vise à répondre aux besoins de ses clients en leur offrant une application facile à gérer afin de les fidéliser

1. Identification des acteurs :

Dans le but d’interagir avec le système, nous avons dégagé les acteurs suivants:

● L’administrateur

● Les clients : ce sont les personnes qui bénéficieront des services de l’application.

● Le docteur : responsable au traitement du cas d’un patient.

● Utilisateur : qui correspond à la personne non inscrite voulant consulter le catalogue. Chaque acteur possède un espace utilisateur (sauf un utilisateur non inscrit).

4. Spécification des besoins fonctionnels et non fonctionnels :

4.1. Les besoins fonctionnels :

Les besoins fonctionnels représentent les actions que le système doit exécuter. Cette application doit couvrir principalement les besoins fonctionnels suivant cités par ordre de priorité :

1. Gestion des utilisateurs : Un administrateur peut ajouter, modifier, supprimer et consulté les listes des clients et docteurs.
2. Gestion avis : le client ou le docteur peut ajouter un commentaire ou une réclamation aussi ils peuvent modifier ou supprimer un commentaire.
3. Gestion RDV : un patient peut ajouter un rendez-vous le modifier ou le supprimer.
4. Gestion formulaire : le patient peut remplir le formulaire ou le modifier selon leurs symptômes.
5. Gestion authentification : un admin peut remplir les coordonnées du client et confirmer son compte ou l’annulée aussi le client a le droit d’ajouter individuellement son propre compte.
6. Gestion évènement : l’admin peut ajouter un évènement spécial tel que l’apparition d’une épidémie ou d’un virus de l’année courante ou bien le supprimer ou le modifier.

4.2. Les besoins non fonctionnels :

Notre application doit répondre aux critères suivants :

● Authentification : chaque acteur peut s'accéder à son espace spécifique préalablement mis, en s'indentification par des informations personnelles.

● L'ergonomie : l'application offre une interface conviviale et facile à utiliser.