

SG01 – Makair

MAINDRON Arthur

Dossier technique du projet – partie individuelle

I - SITUATION DANS LE PROJET	2
I.1 - INTRODUCTION	2
I.2 - SYNOPTIQUE DE LA REALISATION	5
II - REALISATION DE LA « BASE DE DONNEES »	7
II.1 - CONCEPTION DETAILLEE	7
II.2 – TEST(S) UNITAIRE(S)	7
II.1.1 - Test unitaire du module matériel/logiciel.....	10
II.1.2 - Problèmes rencontrés	10
III - REALISATION DE LA FONCTION « PORTAIL »	11
II.1 – CONCEPTION DETAILLEE	11
II.2 - TEST(S) UNITAIRE(S)	12
III.1.1 - Test unitaire du module matériel/logiciel.....	14
III.1.2 - Problèmes rencontrés	14
IV - BILAN DE LA REALISATION PERSONNELLE	15

I - Situation dans le projet

I.1 - Introduction

MakAir est utilisé dans un contexte hospitalier. Dans ce contexte, plusieurs patients peuvent être sous assistance respiratoire. Il se pose alors un problème de surveillance de plusieurs patients simultanément et de la gestion des alarmes associées aux patients. C'est l'objet de cette extension du projet MakAir initial. Permettre la supervision simultanée et à distance d'un certain nombre (non limité et défini par l'utilisateur) de MakAirs.

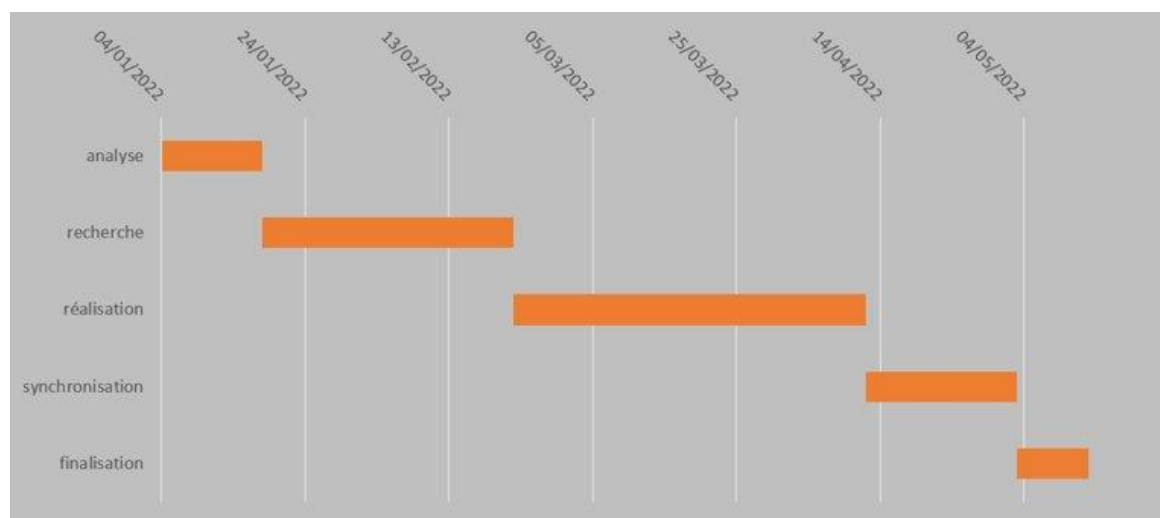
	Fonctions à développer et tâches à effectuer	
Étudiant 3 (MAINDRON) <u>XX</u> EC <input type="checkbox"/> IR <input checked="" type="checkbox"/>	Liste des fonctions / UCs assurés par l'étudiant :	Installation : ✓ Installation de la base de Données. Mise en œuvre : ✓ Avec l'étudiant 4, définir la Base de données. (Format des données, format d'une session) ✓ Avec l'étudiant 2, définir le lien avec le UC 3 « Détecter les MakAirs ». ✓ Librairie DataTable Configuration : ✓ Configurer la Base de données. Réalisation : ✓ Définir la/les IHMs associées aux Use Cases ✓ Codage des classes et des IHMs associées aux Use Cases. Documentation : ✓ Procédure de configuration du Centralisateur ✓ Dictionnaire de classes
	Centralisateur <ul style="list-style-type: none"> ▪ UCC2 – Configurer session supervision ▪ UCC4 – Ajouter un MakAir ▪ UCC5 – Gérer les alarmes 	

Nom du cas d'utilisation	UCC2 – Configurer session de supervision
Pré-condition(s)	Les MakAirs sont disponibles sur le réseau et prêts à être détectés.
Scénario nominal	Le Use Case scrute le réseau à la recherche de MakAirs disponibles. L'agent hospitalier définit les alarmes.
Séquencement	Scruter le réseau à la recherche de MakAirs à enregistrer dans la session. Extension, il peut se faire qu'un MakAir soit à ajouter à la session de supervision. Si un MakAir est nouveau sur le réseau il peut être associé à une session déjà existante. Définir les alarmes
Post-condition	Tous les MakAirs du réseau sont détectés et enregistrés comme faisant partie de la session. Les informations sont archivées dans la base de données
Exigences	

Nom du cas d'utilisation	UCC4 – Ajouter un MakAir
Pré-condition(s)	Les MakAirs sont disponibles sur le réseau et prêts à être détectés, ainsi que le nouveau MakAir que l'on souhaite joindre à la session.
Scénario nominal	Le Use Case détecte les MakAirs en scrutant le réseau
Séquencement	Le centralisateur affiche les MakAirs de la session ainsi que ceux disponibles sur le réseau. Le nouveau MakAir apparaît par différence. Il peut alors être ajouté à la session.
Post-condition	Tous les MakAirs du réseau sont listés et la session est définie
Exigences	

Nom du cas d'utilisation	UCC5 – Gérer les alarmes
Pré-condition(s)	Les données sont disponibles ainsi que les niveaux d'alarmes.
Scénario nominal	Le Use Case gère les alarmes en fonction des données qui arrivent en temps réel. (période T)
Séquencement	Lorsque nécessaire, les alarmes sont déclenchées et l'Agent hospitalier superviseur est informé par sms.
Post-condition	L'alerte est émise vers l'agent hospitalier
Exigences	

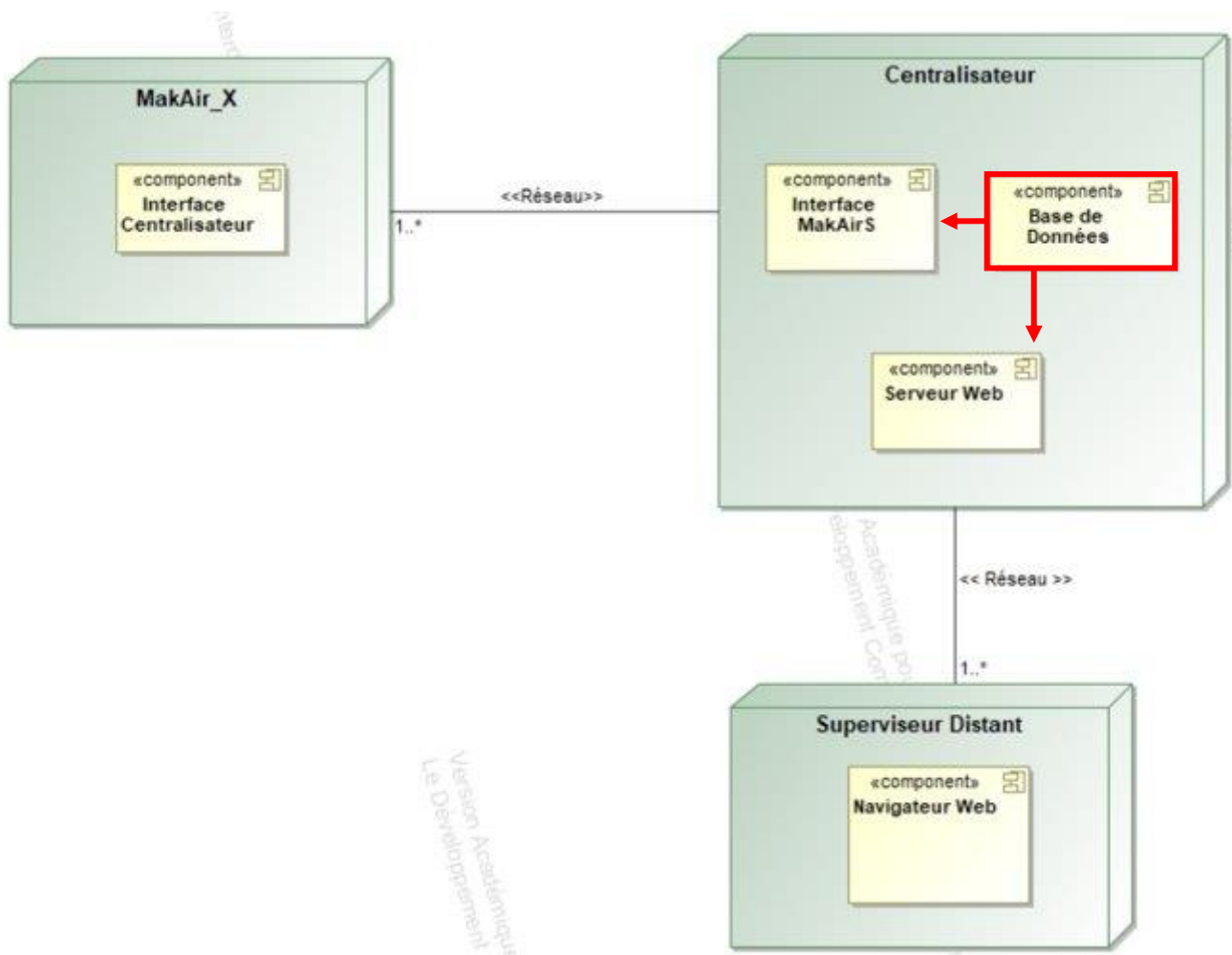
D'après le cahier des charges donné en début de projet, mon travail se tournera principalement autour de la base de données (configuration, prise en charge, etc...) et les données y entrant et sortant. Avec la définition de mon travail grâce au « Cas d'utilisation » 2, 4 et 5. Ces cas d'utilisations définissent et englobent tous les objectifs à atteindre. La suite de mon travail se basera sur la création d'un portail faisant office de pont entre les données d'un patient sur le MakAir (utilisation d'un graphique) et les identifiants d'un personnel soignant.

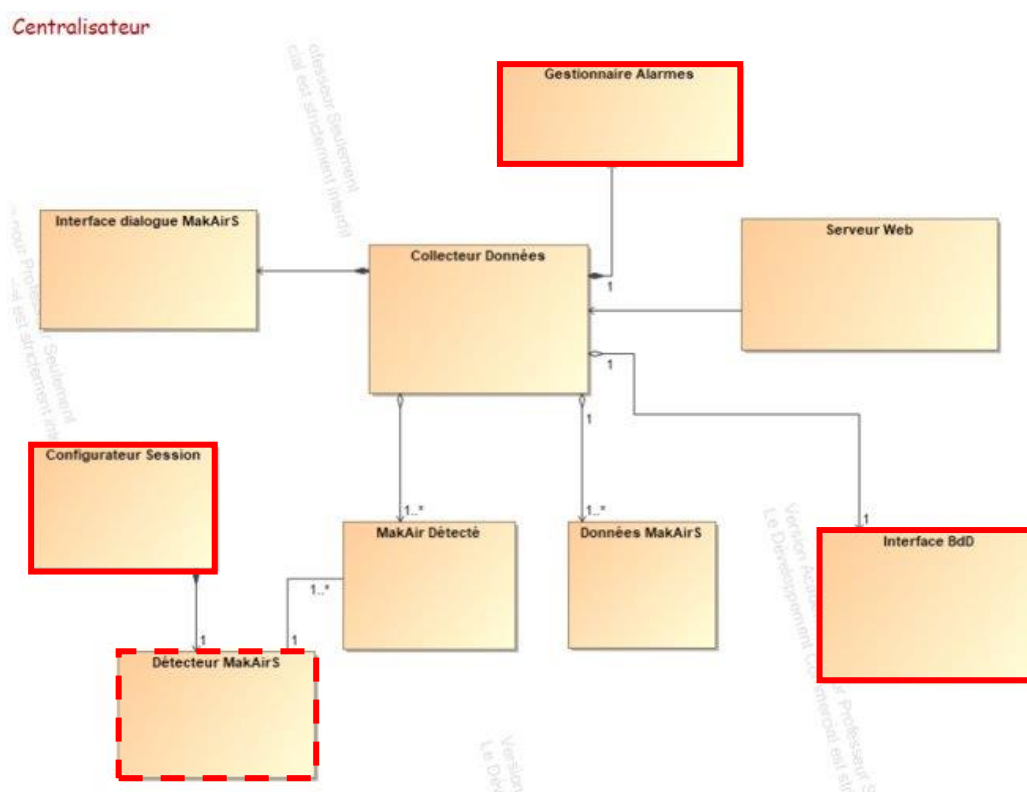


étapes	Date de début	Durée	Date de fin
analyse	04/01/2022	14	18/01/2022
recherche	18/01/2022	35	22/02/2022
réalisation	22/02/2022	58	21/04/2022
synchronisation	21/04/2022	21	12/05/2022
finalisation	12/05/2022	10	22/05/2022

Lors de l'avancement du projet la phase d'analyse dura deux semaines et me permis de définir le format des données de ma BdD pour commencer. Par la suite, lors de cette phase, j'ai pu me mettre en accord avec mes collègues sur certaines tâches complémentaires à réaliser comme le portail ou la conception d'alarmes lorsqu'un certain seuil du graphique d'un patient devient critique.

I.2 - Synoptique de la réalisation

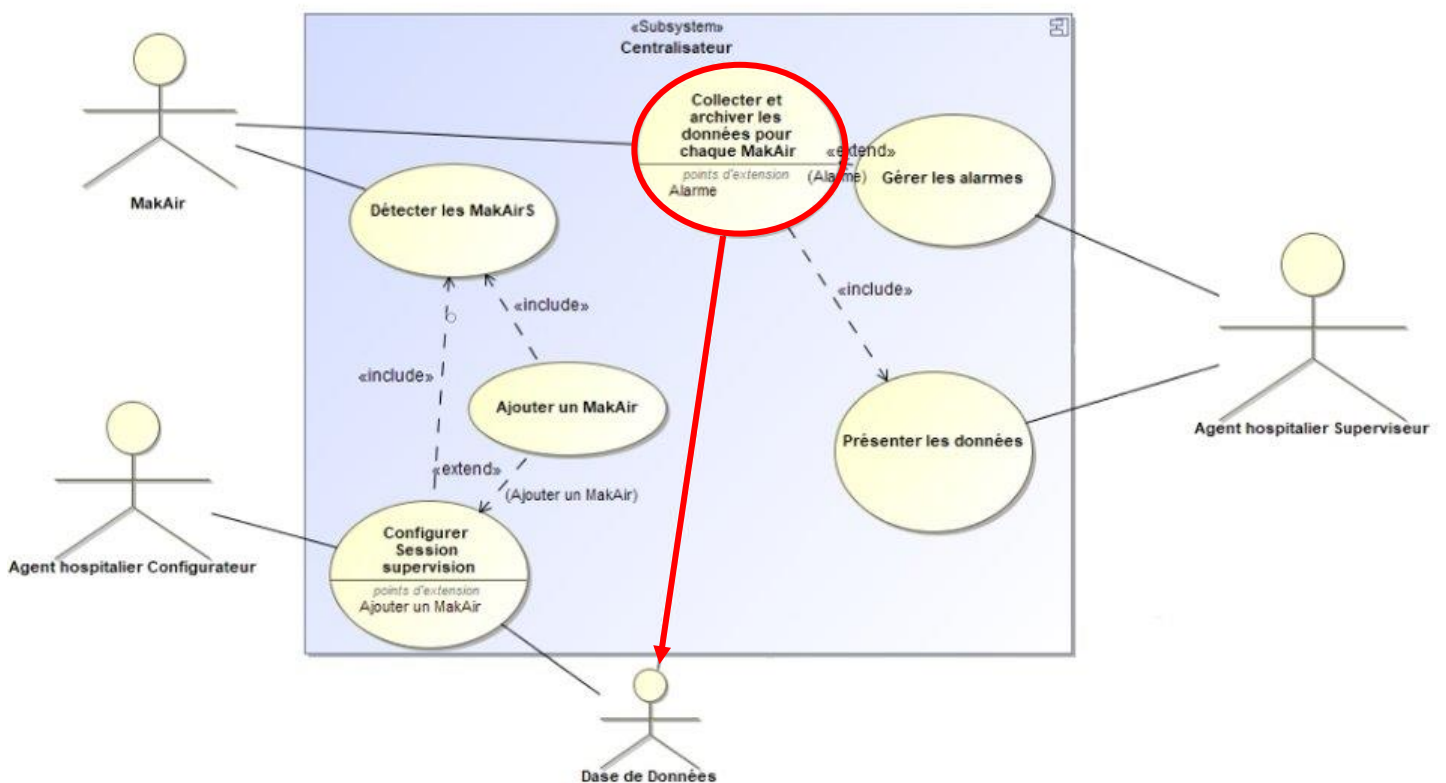




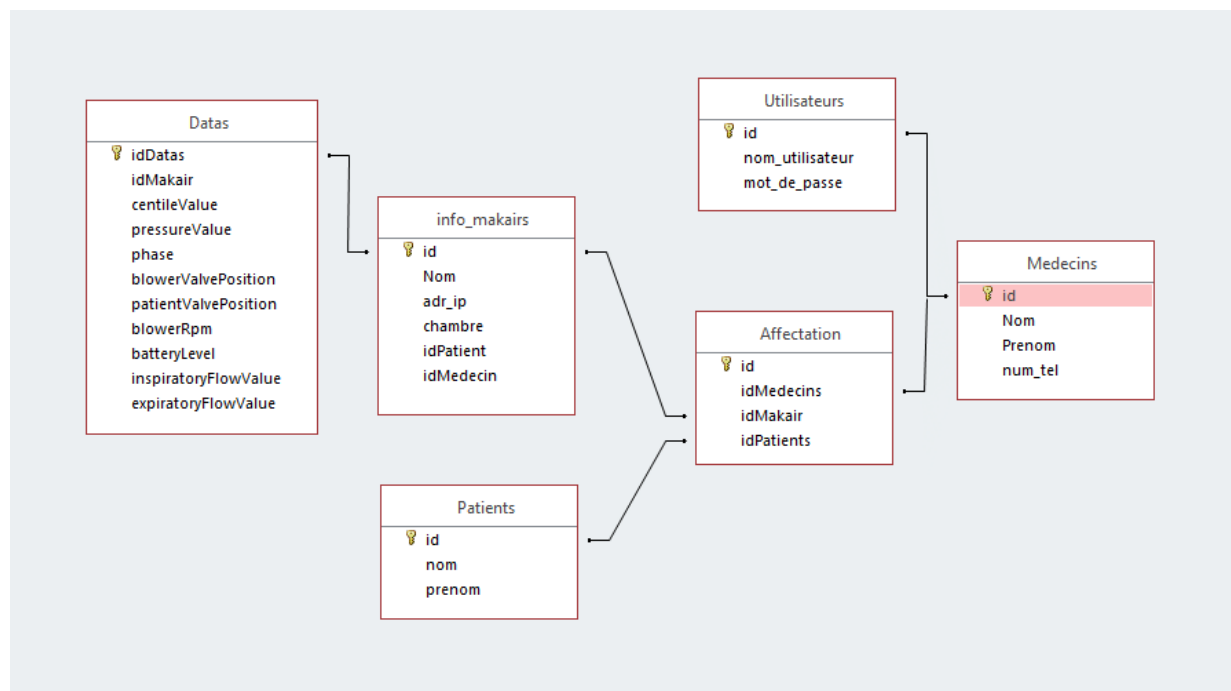
Au sein de ce projet, le diagramme de classe candidate « Centralisateur » nous montre différente partie du travail à mener. Tout cela tournant autour de la collecte des données pour sa réutilisation pour de graphiques ou autre valeurs réutilisés par le personnel soignant.

II - Réalisation de la « base de données »

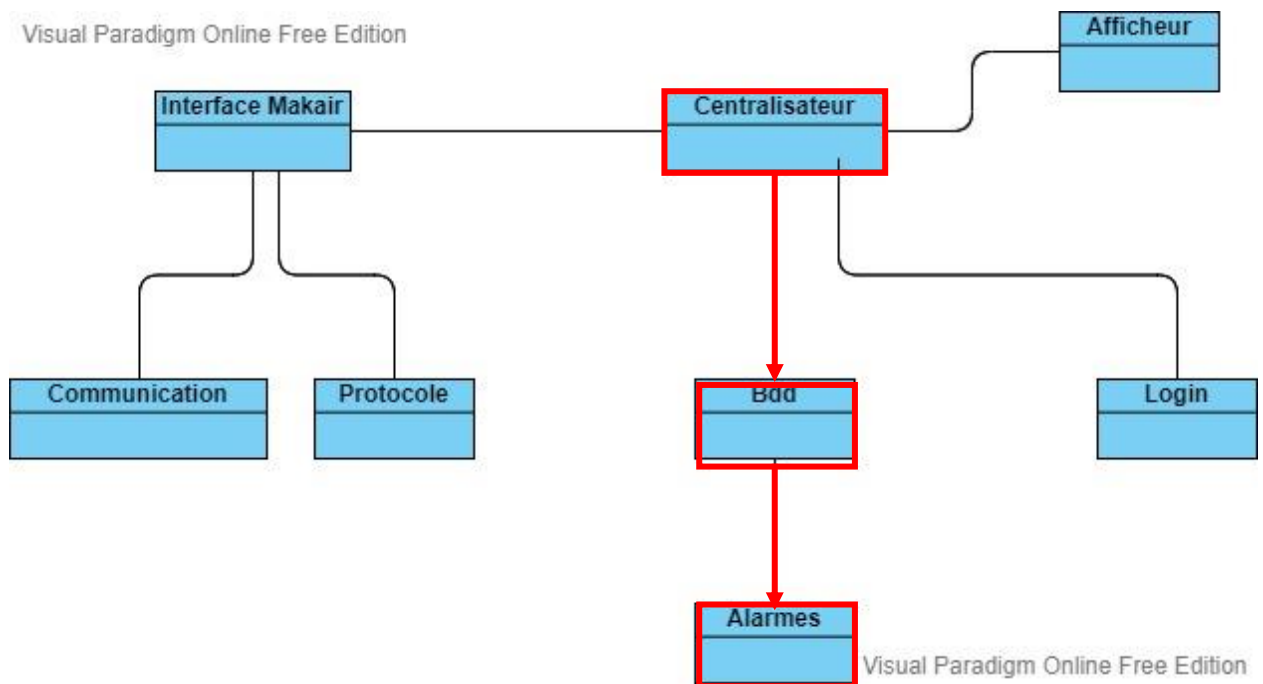
II.1 - Conception détaillée



Ci-dessous la base de données dans laquelle les données du MakAir seront stockées dans « Datas » et celles des patients, des médecins ou de l'utilisateur joints à une table « Affectation », qui fait lien entre chaque patient sur un MakAir donnée et quelle soignant consulte sa courbe respiratoire sur internet.



II.2 – Test(s) unitaire(s)

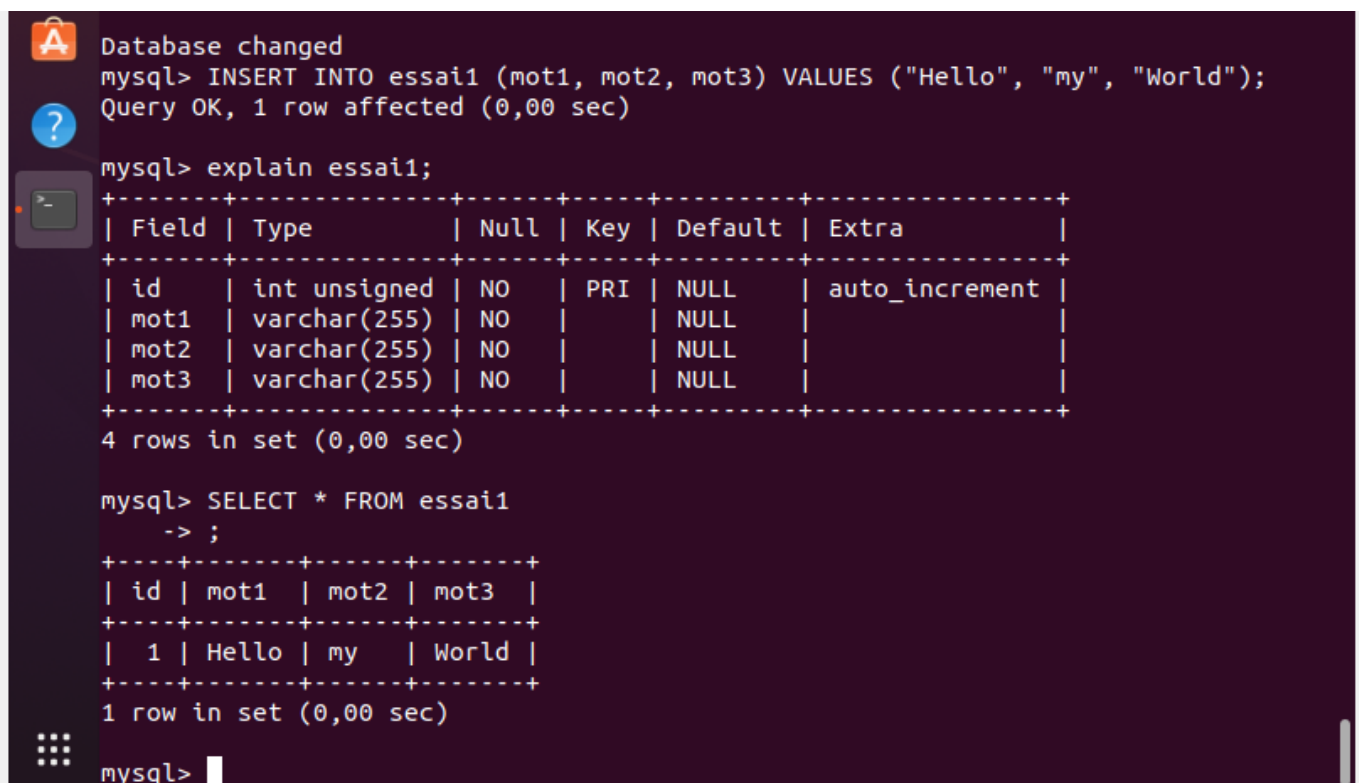


Pour que le Centralisateur puisse fonctionner dans son intégralité il nécessite de communiquer avec la Bdd. Sachant que les alarmes communiqueront à la Bdd de potentiels problèmes dû au patient.


```
mysql> use essai;
Database changed
mysql> drop table if exists table;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that c
orresponds to your MySQL server version for the right syntax to use near 'table'
at line 1
mysql> DROP TABLE IF EXISTS table1
-> ;
Query OK, 0 rows affected, 1 warning (0,00 sec)

mysql> CREATE TABLE essai1
-> ;
ERROR 4028 (HY000): A table must have at least one visible column.
mysql> CREATE TABLE essai1(id int unsigned NOT NULL auto_increment,
-> # Identifiant unique
-> mot1 varchar(255) NOT NULL,
-> mot2 varchar(255) NOT NULL,
-> mot3 varchar(255) NOT NULL, PRIMARY KEY (id));
Query OK, 0 rows affected (0,03 sec)
```

Création de la table « essai1 » faisant office de test avec comme colonnes « mot 1, 2 et 3 » qui stockeront les valeurs « Hello », « my » et « world ». Par la suite différentes tables et valeurs seront définies.



```
Database changed
mysql> INSERT INTO essai1 (mot1, mot2, mot3) VALUES ("Hello", "my", "World");
Query OK, 1 row affected (0,00 sec)

mysql> explain essai1;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id    | int unsigned  | NO   | PRI | NULL    | auto_increment |
| mot1  | varchar(255)  | NO   |     | NULL    |                |
| mot2  | varchar(255)  | NO   |     | NULL    |                |
| mot3  | varchar(255)  | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0,00 sec)

mysql> SELECT * FROM essai1
-> ;
+-----+-----+-----+-----+
| id | mot1 | mot2 | mot3 |
+-----+-----+-----+-----+
| 1  | Hello | my   | World |
+-----+-----+-----+-----+
1 row in set (0,00 sec)

mysql> 
```

II.1.1 - Test unitaire du module matériel/logiciel

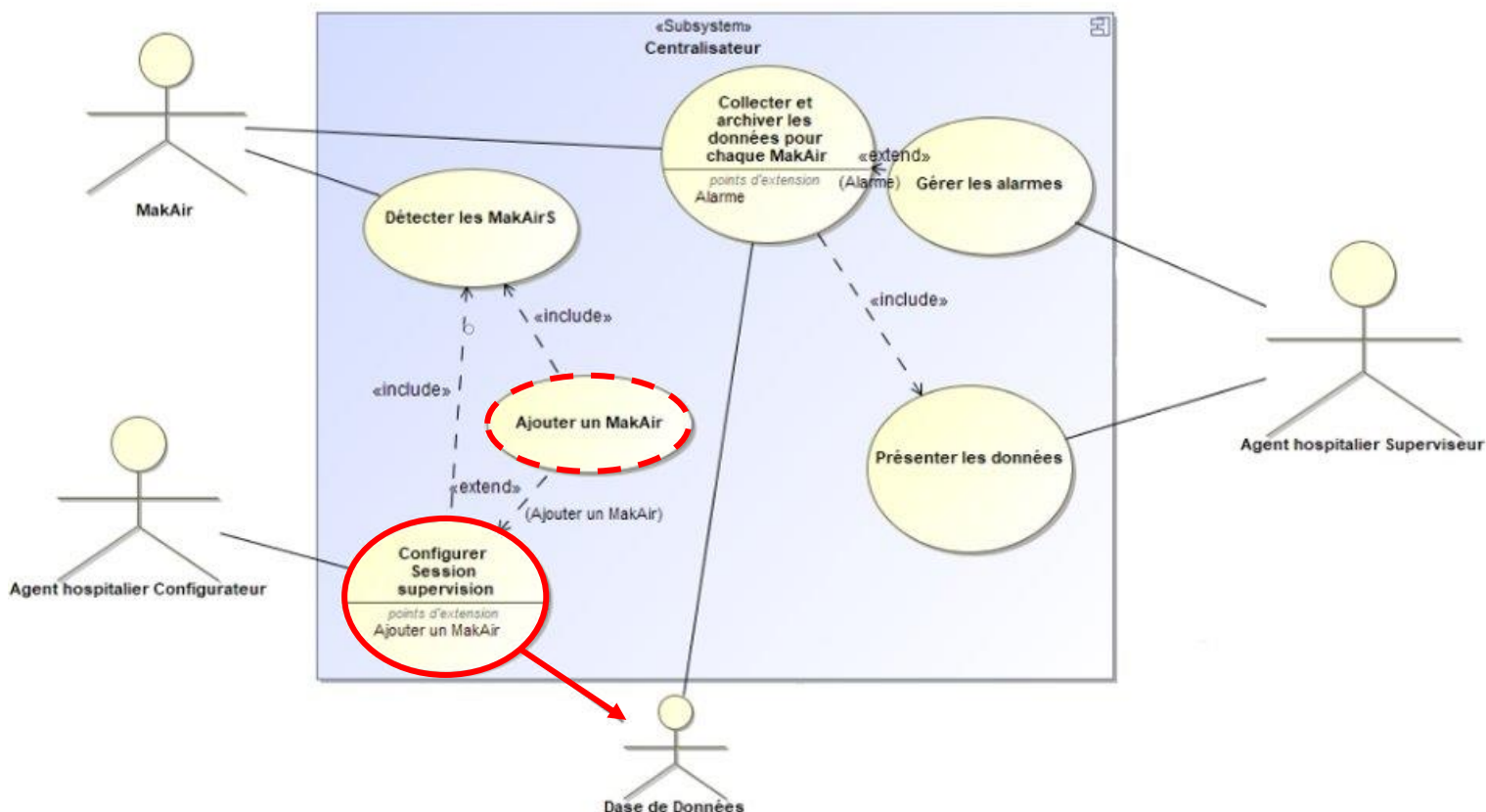
Élément testé :	Base de donnée			
Objectif du test :	Requête SQL			
Nom du testeur :	Maindron Arthur		Date :	28/02/2022
Moyens mis en œuvre :	Logiciel : Ligne de commande LINUX	Matériel : SQL		Outil de développement :
Procédure du test :				
Id	Description du vecteur de test	Résultat attendu	Résultat obtenu	Validation (O/N)
1	Création de la table 'test'	Affichage en ligne de commande	Table 'test' bien créée	O
2	Création des valeurs	Valeurs 'hello my world' dans certaines colonnes d'une table	Valeur bien affichées	O
Conclusion du test :	Création de table optimales créées et fonctionnelles			

II.1.2 - Problèmes rencontrés

Lors de ma phase de test pour la base de données le principal problème que j'ai pu rencontrer est celui de bien pouvoir récupérer les données du MakAir et de pouvoir les accorder dans les bonnes colonnes. Tout cela fût en accord avec le code de Johan Laurent. Qui par la suite seront réutilisé par le code de Paul Gaborit pour être affiché sous forme de graphique dynamique. Ces données seront donc affichées en temps réel.

III - Réalisation de la fonction « portail »

II.1 – Conception détaillée



Pour la création du portail, il est nécessaire d'utiliser une table dans la base de données intitulée « Utilisateurs » qui stockera alors les mots de passes et identifiants de chaque personnel soignant souhaitant consulter les données du patient en temps réel et tout cela à distance.


II.2 - Test(s) unitaire(s)

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <link rel="stylesheet" href="style.css" />
5   </head>
6
7   <body>
8     <div class="container">
9
10
11       <div class="row">
12         <h1>Connexion</h1>
13
14         <hr>
15
16         <form action="verification.php" method="POST">
17
18           <div class="info"><b>Nom d'utilisateur</b></div>
19           <input type="text" placeholder="Entrer le nom d'utilisateur" name="username" required>
20
21           <div class="info"><b>Mot de passe</b></div>
22           <input type="password" placeholder="Entrer le mot de passe" name="password" required>
23         </div>
24
25         <input type="submit" id='submit' value='Connexion' >
26
27
```

Pour pouvoir créer notre portail, l'utilisation du code sous HTML et CSS ou encore PHP est nécessaire. Ci-dessus on peut voir la conception du code pour la page « login.php » qui comporte deux case essentiel intitulées « username » et « password » qui seront directement reliées avec la base de données pour pouvoir récupérer les valeurs correspondantes dans la table

```
1 <?php
2 session_start();
3 if(isset($_POST['username']) && isset($_POST['password']))
4 {
5     // connexion à la base de données
6     $db_username = 'makair';
7     $db_password = 'makair';
8     $db_name     = 'makair';
9     $db_host     = '192.168.115.101';
10    $db = mysqli_connect($db_host, $db_username, $db_password, $db_name)
11        or die('could not connect to database');
12
13    // on applique les deux fonctions mysqli_real_escape_string et htmlspecialchars
14    // pour éliminer toute attaque de type injection SQL et XSS
15    $username = mysqli_real_escape_string($db, htmlspecialchars($_POST['username']));
16    $password = mysqli_real_escape_string($db, htmlspecialchars($_POST['password']));
17
18    if($username !== "" && $password !== "")
19    {
20        $requete = "SELECT count(*) FROM Utilisateurs where
21                    nom_utilisateur = '". $username. "' and mot_de_passe = '". $password. "' ";
22        $exec_requete = mysqli_query($db, $requete);
23        $reponse      = mysqli_fetch_array($exec_requete);
24        $count = $reponse['count(*)'];
```

Dans la partie de code ci-dessus, une requête PHP est utilisée, et cherche à se connecter tout d'abord à la base de donnée grâce à un identifiant et un mot de passe pour s'y connecter (on mettra les mêmes noms par soucis de simplicité). Une fois la connexion établie on souhaite chercher dans la table « Utilisateurs » si le mot de passe et l'identifiant correspond bien à ce que l'on trouve en base de données.



Ci-dessus le résultat du code « login.php »

III.1.1 - Test unitaire du module matériel/logiciel

Élément testé :		Code HTML		
Objectif du test :		Portail de connexion fonctionnel		
Nom du testeur :		Maindron Arthur	Date :	30/02/2022
Moyens mis en œuvre :		Logiciel : Format texte, codage HTML/PHP	Matériel : HTML/PHP	Outil de développement :
Procédure du test :				
Id	Description du vecteur de test	Résultat attendu	Résultat obtenu	Validation (O/N)
1	Création de la page 'portail' grâce au réseau interne privée intitulé « réseau TP »	Affichage d'un portail avec « login » plus « password »	Page bien créée	O
2	Création de la synchronisation entre le portail et les valeurs d'un graphique	Valeurs 'hello my world' dans certaines colonnes d'une table	Valeur bien affichées	O
Conclusion du test :		Création de table optimales créées et fonctionnelles		

III.1.2 - Problèmes rencontrés

Le principal problème rencontré fût la synchronisation entre les valeurs de la base de données prise par le code de Johan Laurent et réutiliser ces valeurs dans le tableau graphique de Paul Gaborit. Nous avons dû travailler à trois durant plusieurs heures sur ce code.

IV - Bilan de la réalisation personnelle

Durant la conception active du projet j'ai pu réaliser deux tâches principales et essentiel au groupe par la suite. Peu de problèmes furent rencontrés et s'ils y en avaient ils étaient tous résolu dans la semaine ou bien les jours suivants. Le projet nécessite encore quelques améliorations notamment sur les alarmes, indices important du graphique permettant au personnel soignant d'avoir un réel suivi sur un patient donné. Cette partie du projet fût, pour le moment, réfléchi mais pas abouti. Le projet possède un potentiel d'intérêt plus important que d'autre puisqu'il est plus concret, à mon sens. Le projet fût développé il n'y a pas si longtemps et devient presque nécessaire pour les soignants aujourd'hui. De plus peu de temps avant notre revue finale de ce projet, une audition au près du CHU de Cholet sera organisé pour la présentation de notre projet dans sa globalité. L'intérêt de ce projet est alors bien plus intéressant qu'un autre puisqu'il suscite l'attention du domaine professionnel et surtout médicale qui devient de nos jour une priorité.