Projet RespireStat

SLAM 1e année

[Sujet 2](#_Toc65702940)

[Les données 2](#_Toc65702941)

[Les fonctionnalités 2](#_Toc65702942)

[Les modifications à effectuer 2](#_Toc65702943)

[Maquettes intermédiaires 3](#_Toc65702944)

[Phase de maquettage 6](#_Toc65702945)

[Phase de développement 6](#_Toc65702946)

[Documents à rendre 6](#_Toc65702947)

[Technologies utilisées 7](#_Toc65702948)

[Interface graphique avec Swing 7](#_Toc65702949)

[Gestionnaire de version GIT 7](#_Toc65702950)

[Gestionnaire de dépendance Maven 7](#_Toc65702951)

[Importer et lire un fichier CSV avec openCSV 7](#_Toc65702952)

[Architecture technique du logiciel lourd 8](#_Toc65702953)

[Package model 8](#_Toc65702954)

[Package vue 8](#_Toc65702955)

[Package controller 8](#_Toc65702956)

# Sujet

Datactivist est une société coopérative et participative qui se donne pour mission d’ouvrir les données et de les rendre utiles et utilisées.

StatPerf est une ESN spécialisée dans le traitement des données statistiques et la visualisation de données. Vous êtes développeur dans leur équipe logicielle.

Respire est une association nationale pour l’amélioration et la préservation de la qualité de l’air.

Respire a fait appel à Datactivist pour collecter les données sur la pollution aérienne aux abords des écoles et crèches d’Île-de-France afin d'alerter sur les niveaux d'expositions des populations fragiles à la pollution aérienne.

Respire fait appel StatPerf pour concevoir et réaliser une application lourde nommée RespireStat permettant de visualiser les données collectées par Datactivist.

Vous êtes développeur salarié au sein de StatPerf, plus spécifiquement dans le département développement logiciel.

Il vous a été confié la mission d’améliorer ce logiciel qui est incomplet.

## Les données

Les données de Datactivist sur le sujet sont en accès libre sur le site du gouvernement français : <https://www.data.gouv.fr/fr/datasets/base-de-donnees-de-la-pollution-aerienne-aux-abords-des-ecoles-et-creches-dile-de-france/>

Le fichier en format CSV est déjà dans le dossier compressé RespireStat.zip qui vous a été fourni.

## Les fonctionnalités

Les fonctionnalités du logiciel une fois terminé seront les suivantes :

* **F01** : Pour chaque polluant (NO2, PM10, PM25), montrer l’établissement pour lequel le taux d’exposition est le plus élevé pour chaque année, de 2012 à 2017 : préciser sa ville et sa distance à Paris (prendre le centre 0 des km comme centre de Paris) en utilisant ses coordonnées GPS.

Exemple simplifié (avec données factices) :

|  |  |  |
| --- | --- | --- |
| NO2 | PM10 | PM25 |
| Collège Saint Martin  Paris 13  Distance : 5km | Lycée Baudelaire  Courbevoie  Distance : 15 km | École élémentaire  Ivry  Distance : 7km |

* **F02** : Pour chaque ville, donnez sa moyenne pour chaque polluant en 2017. Trier les villes de la plus polluée à la moins polluée. Un arrondissement est considéré comme une ville.
* **F03** : même fonctionnalité que F02 mais pour les départements.
* **F04** : Pour chaque département, donnez le pourcentage d’évolution (en moyenne de tous les établissements) de chaque polluant entre chaque année de 2012 à 2017

## Les modifications à effectuer

Vous travaillez pour l’entreprise de prestation chargée du développement de ce logiciel. Vous êtes missionné pour finir le logiciel. Quelques fonctionnalités n’ont pas encore été développées, et d’autres sont en cours.

F01 : **ECRAN N°2** Pour cette première fonctionnalité, elle est presque complète. Il manque la distance à Paris. *Vous devrez vous renseigner sur le calcul de la distance d’une coordonnée GPS à une autre*.

F02 : **ECRAN N°3** Cette fonctionnalité est mise en place et ne comporte pas de bugs. Vous vous baserez sur celle-ci pour comprendre comment doivent être développés les autres écrans.

F03 : **ECRAN N°4** Cette fonctionnalité est en cours de développement. Vous devez la finir.

F04 : **ECRAN N°5** Cette fonctionnalité n’a pas été mise en place, vous devez l’implémenter sur un écran N°5.

### Maquettes intermédiaires

Une image contenant texte

Description générée automatiquement

Figure 1 Ecran N°0a

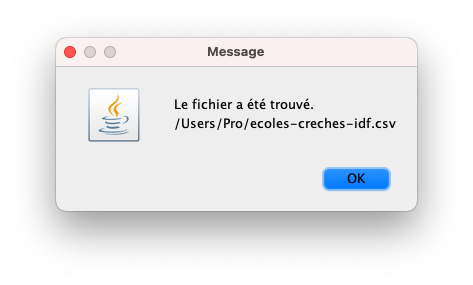


Figure 2 Ecran N°0b

Une image contenant texte

Description générée automatiquement

Figure 3 Ecran N°1

Une image contenant table

Description générée automatiquement

Figure 4 Ecran N°2

Une image contenant table

Description générée automatiquement

Figure 5 Ecran N°3

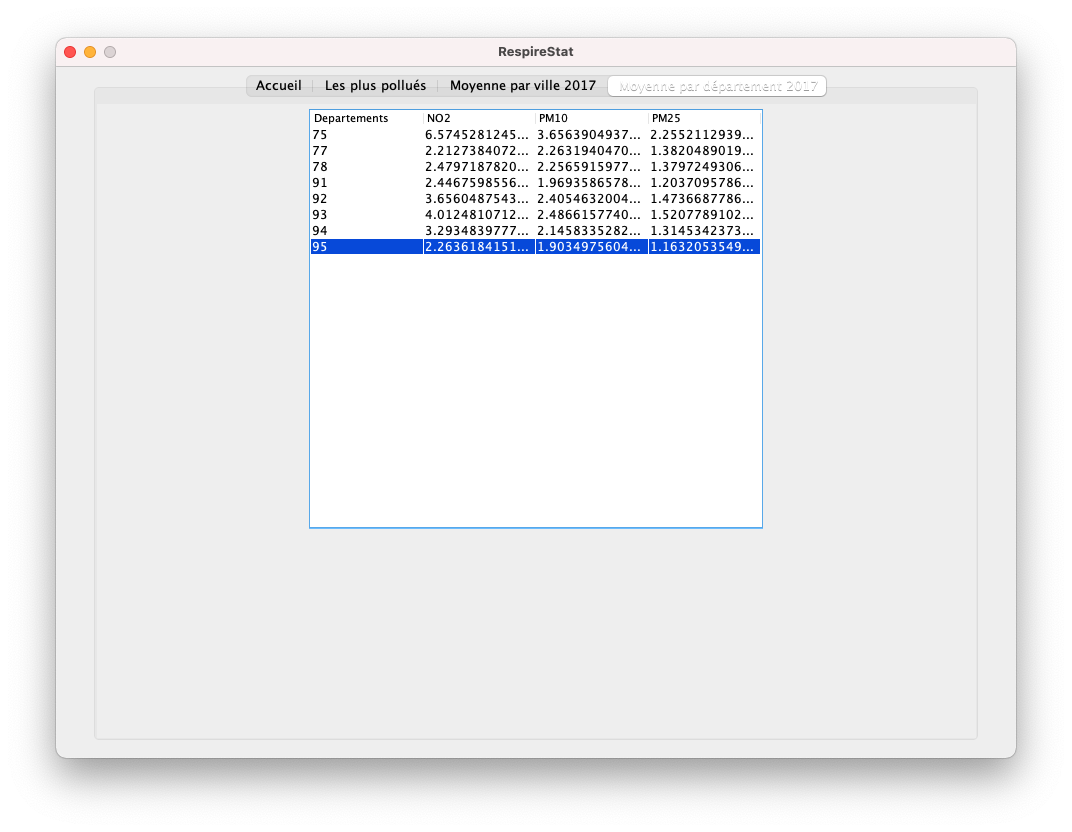


Figure 6 Ecran N°4

## Phase de maquettage

1. Vous devrez modifier la maquette de l’écran N°2 afin d’intégrer la distance.

|  |  |  |  |
| --- | --- | --- | --- |
| 2012 | NO2 | PM10 | PM25 |
| Identifiant | 8629 | 8629 | 8629 |
| Nom | Crèche… | Crèche… | Crèche… |
| Ville | Paris 10 | Paris 10 | Paris 10 |
| Département | 75 | 75 | 75 |
| Taux | 110.01384 | 110.01384 | 110.01384 |
| Distance ( de Paris ) | 0km | 0km | 0km |

1. Vous devrez proposer une maquette pour l’écran N°5 présentant la fonctionnalité F04.

Une image contenant texte, capture d’écran, ligne, Police

Description générée automatiquement

## Phase de développement

1. Créez un repository GitLab dont le nom sera RespireStatG1 si vous êtes le groupe 1, etc
2. Créez des issues dans GitLab pour chaque fonctionnalité à implémenter
3. Découvrez le code existant ensemble, essayez de comprendre son fonctionnement !

## Documents à rendre

1. Les maquettes.
2. Un manuel d’utilisation du logiciel en français et en anglais.
3. Le code source grâce à un lien GitLab
4. L’exécutable .jar (Générer un .jar sur Eclipse : File>Export>Runnable JAR File)

# Technologies utilisées

Le dossier compressé RespireStat.zip contient le projet à importer directement dans Eclipse. En principe, vous n’avez pas besoin de tous les éléments suivants. Je les mets cependant à votre disposition ils auraient étaient indispensables à la création du projet.

## Interface graphique avec Swing

Vous utiliserez la bibliothèque **Swing** incluse dans Java.

[Cliquez ici pour accéder au tutoriel de base de Swing](https://baptiste-wicht.developpez.com/tutoriels/java/swing/debutant/)

## Gestionnaire de version GIT

Vous devrez utiliser le gestionnaire de version **GIT**.

Une branche doit être créée pour la version stable et une autre pour la version en cours de développement.

Les accès doivent être communiqués au professeur pour qu’il puisse cloner votre projet.

Vous pouvez utiliser Git directement dans l’IDE Eclipse. Pour cela, faites un clic droit sur le projet et cherchez « Team » puis « Share Project ».

## Gestionnaire de dépendance Maven

**Maven** est un gestionnaire de dépendances que nous utiliserons afin d’importer opencsv.

Pour installer Maven sur Eclipse :

1. Open Eclipse IDE
2. Click Help -> Install New Software...
3. Click Add button at top right corner
4. At pop up: fill up Name as "M2Eclipse" and Location as <http://download.eclipse.org/technology/m2e/releases>

Puis transformez votre projet Java en un projet Maven (Clic droit sur le projet puis « Configure » puis « Convert to Maven Projet ».

## Importer et lire un fichier CSV avec openCSV

**Opencsv** vous permettra de charger et lire le fichier csv. Vous pouvez consulter la documentation à l’adresse suivante : <https://docs.opencv.org/4.0.0/d1/d0a/tutorial_java_eclipse.html>

Après avoir installé Maven ajoutez ces lignes dans le fichier pom.xml :

<dependencies>

<dependency>

<groupId>com.opencsv</groupId>

<artifactId>opencsv</artifactId>

<version>4.0</version>

</dependency>

</dependencies>

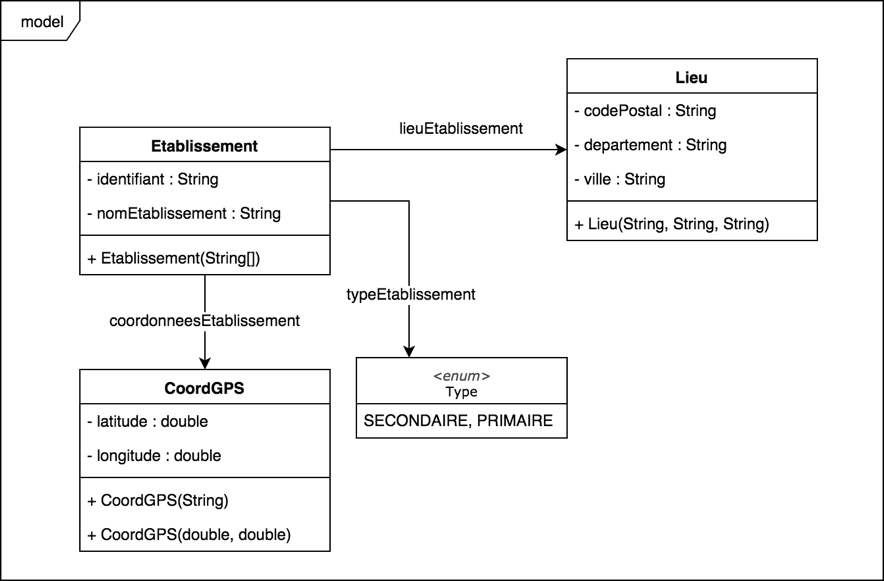
*Sur le projet qui vous est fourni, ces manipulations ont déjà été effectuées*

# Architecture technique du logiciel lourd

Le projet Java est organisé en 3 packages :

* model (contiendra les classes du modèle de données)
* vue (contiendra les classes liées à l’interface graphique)
* controller (contiendra les classes effectuant des traitements)

## Package model



Ci-dessus vous trouverez le diagramme de classes minimal du package « model » uniquement. Vous devrez bien entendu créer d’autres variables, constructeurs et méthodes mais cette base doit être respectée.

## Package vue

Vous êtes libre d’utiliser les éléments Swing et le graphisme de votre choix.

Conseil : vous pouvez utiliser des onglets grâce à *JTabbedPane* et des tableaux grâce à *JTable*, *JScrollPane* et *AbstractTableModel*.

## Package controller

**ConvertCSV.java**

Cette classe Java ne contient que des méthodes et des variables de classe. C’est elle qui va lire le fichier CSV et stocker les informations dans le modèle.

**StatEtab.java**

Cette classe ne contiendra que des méthodes de classe destinées à effectuer des statistiques sur la liste des établissements