

Enseignant(s)

**SANANES Frédéric**

Email(s)

[sananes@myges.fr](mailto:sananes@myges.fr)

## Next Door Buddy (3AL1)

### 1 Matières, formations et groupes

Matière liée au projet :

Formations : -

Nombre d'étudiant  
par groupe :

**2 à 3**

Règles de constitution des groupes: **Libre**

Charge de travail  
estimée par étudiant : **70,00 h**

### 2 Sujet(s) du projet

Type de sujet : **Sujet libre à valider**

### 3 Détails du projet

#### Objectif du projet (à la fin du projet les étudiants sauront réaliser un...)

Construire une application extensible (API et clients divers) sur le thème "les copains du quartier"

Le projet devra proposer des services utilisables par les habitants d'un quartier :

- troc entre voisins
- services (récupérer un colis, sortir le chien, faire les courses d'une personne âgée ...)
- sorties entre voisins : par exemple, une personne propose d'aller au cinéma avec au maximum cinq autres voisins, à une certaine date, qui doivent s'inscrire pour que l'activité ait lieu, sinon elle est annulée
- surveillance en cas d'absences
- messagerie (un plus serait d'intégrer la vidéo), servant également à savoir qui est connecté ou ne l'est pas ...
- organisation d'événements communautaires (par exemple, nettoyage, collecte de déchets ...)
- jeux
- et un journal de quartier dont les pages seront stockées en MongoDB

Il sera important de mémoriser les différentes interactions entre voisins du quartier (par exemple, A a sorti le chien de B, B est allé au cinéma avec C etc ...), notamment pour proposer des suggestions de rencontre.

Noter que l'application Web construite devra être conteneurisée.

Cette liste n'est pas exhaustive et peut être personnalisée.

## Descriptif détaillé

Le projet doit proposer impérativement :

- une définition de services WEB (définis en NodeJS) donnant accès à un mécanisme de persistance évolutif (SGBD, ...)
- un client Web React à destination des utilisateurs de l'application et un autre pour assurer le back office; il intégrera un moteur de suggestion (par exemple, au vu des interactions de B et de D, il serait intéressant de les faire se rencontrer)
- un client lourd Java, permettant d'assurer des missions de webscrapping (spécifications de sites à visiter, récupération et traitement de données, automatisation du processus de recherche) possédant une interface graphique de type JavaFX. Intégrer la possibilité de fixer des catégories d'informations à rechercher.
- des fonctionnalités avancées en Java
  - \* Mettre en place un système de mise à jour du logiciel (pilotee par un serveur)
  - \* Système de thèmes pour customiser l'application (par exemple on peut télécharger des thèmes qui permettent de modifier l'apparence de l'application - changement couleurs, arrangement, polices...)
  - \* Système online/offline
  - \* Un système de plugins (par exemple de l'export de fichier, des outils d'analyse statistiques etc ...)
- un langage d'interrogation (type SQL) pour manipuler des documents (en utilisant lex et yacc).

De plus, cette construction doit s'appuyer sur une vraie démarche de gestion de projet et l'utilisation d'un Trello est obligatoire.

## Ouvrages de référence (livres, articles, revues, sites web...)

## Outils informatiques à installer

IntelliJ, JDK version récente, SGBD (JDBC/MongoDB), NodeJS, React  
L'utilisation de frameworks est libre (sous réserve de validation)  
L'utilisation d'outils cross-platform est interdite

## 4

## Livrables et étapes de suivi

0

## Définition du sujet

### Présentation des règles du projet

- Constitution des trinômes (avant le jeudi 20 février 23h59) et délimitation du type de sujets à construire (avant le dimanche 16 mars 23h59)
- Explication sur l'organisation et les modalités de travail
- Conditions de validation du projet

une définition de services WEB (définis en NodeJS) donnant accès à un mécanisme de persistance évolutif (SGBD, ...)

- un client Web React à destination des utilisateurs de l'application et un autre pour assurer le back office; il intégrera un moteur de suggestion (par exemple, au vu des interactions de B et de D, il serait intéressant de les faire se rencontrer)

- un client lourd Java, permettant d'assurer des missions de webscrapping (spécifications de sites à visiter, récupération et traitement de données, automatisation du processus de recherche) possédant une interface graphique de type JavaFX. Intégrer la possibilité de fixer des catégories d'informations à rechercher.

- des fonctionnalités avancées en Java

\* Mettre en place un système de mise à jour du logiciel (pilote par un serveur)

\* Système de thèmes pour customiser l'application (par exemple on peut télécharger des thèmes qui permettent de modifier l'apparence de l'application - changement couleurs, arrangement, polices...)

\* Système online/offline

\* Un système de plugins (par exemple de l'export de fichier, des outils d'analyse statistiques etc ...)

- un langage d'interrogation (type SQL) pour manipuler des documents (en utilisant lex et yacc).

De plus, cette construction doit s'appuyer sur une vraie démarche de gestion de projet et l'utilisation d'un Trello est obligatoire.

réunion : 19/02

mercredi  
19/02/2025  
23h59

1	Etape intermédiaire	<p>Validation définitive du sujet Il est nécessaire d'avoir posté sur MyGES, avant le 16 mars 2025, 23h59 :</p> <ul style="list-style-type: none"> <li>- la (les) proposition(s) de projet et un bref descriptif fonctionnel (faire une page de présentation)</li> <li>- pour chacune d'entre elles, des idées de traitement</li> <li>- pour chacune d'entre elles, proposer un nom de projet et un logo identifiant l'application (réfèrent visuel)</li> </ul> <p>Rqs :</p> <ul style="list-style-type: none"> <li>- être très précis dans le descriptif de l'application</li> <li>- si le projet contient un jeu basé sur les relations humaines, rendre les règles détaillées de fonctionnement du jeu à la validation</li> </ul> <p>Tous les sources devront être déposé sur un compte Github et identifiés par développeur et un Trello devra être utilisé pour assurer la gestion de projets.</p>	<p><b>dimanche</b> <b>16/03/2025</b> <b>23h59</b></p>
2	Etape intermédiaire	<p>Validation des règles fonctionnelles demandées Amener obligatoirement les premiers programmes construits (premières pages Web, test d'API etc ...); la partie lourde (Java) doit être quasiment terminée</p> <p>Fournir :</p> <ul style="list-style-type: none"> <li>¥ planification de l'application (type quelconque),</li> <li>¥ Documentation technique complète (descriptif fonctionnel ou cahier des charges, use cases, MCD et/ou diagramme de classes, justification architecture logicielle, définition des algorithmes complexes à utiliser)</li> <li>¥ Dessin des IHM souhaitées</li> <li>¥ Description des APIs/Frameworks extérieurs à JSE utilisés dans l'application</li> </ul> <p>réunion : 15/04 3AL1 et 16/04 3AL2</p>	<p><b>dimanche</b> <b>13/04/2025</b> <b>23h59</b></p>
4	Etape intermédiaire	<p>Vérification des documents remis en forme et corrigés Etat d'avancement et vérification de la partie API, questions diverses Démonstration du client lourd finalisé;</p> <p>Fournir :</p> <ul style="list-style-type: none"> <li>¥ Reprendre les dossiers en intégrant les modifications vues pendant la séance précédente, pour finaliser le dossier fonctionnel et technique complet</li> <li>¥ Etat d'avancement (Trello)</li> <li>¥ API terminée</li> <li>¥ client WEB pour utilisateur</li> </ul> <p>réunion : 14/05</p>	<p><b>dimanche</b> <b>11/05/2025</b> <b>23h59</b></p>

5	Etape intermédiaire	<p>Démonstration du client web et au moins 95 % de l'application totale réalisé</p> <p>Fournir :</p> <ul style="list-style-type: none"> <li>¥ Reprendre les dossiers en intégrant les modifications vues pendant la séance précédente, pour finaliser le dossier fonctionnel et technique complet</li> <li>¥ Etat d'avancement (Trello)</li> <li>¥ Client Web React administration terminée</li> <li>¥ Micro-langage</li> </ul> <p>réunion : 30/06 3AL1 et 01/07 3AL2</p>	<p><b>dimanche</b>  <b>29/06/2025</b>  <b>23h59</b></p>
6	Rendu final	<p>Intégralité des sources (nettoyés des traces de débogage), des classes et des exécutables de l'application (si existants)</p> <p>Intégralité des fichiers du front/back : html, javascript et Node.js, multimedia (images, vidéo, sons) + driver BDD</p> <p>Base de données sous la forme d'un fichier texte à importer (prévoir plusieurs jeux d'essais, dont un vide)</p> <p>Dossier technique complet et dossier utilisation sur plateforme,</p> <p>Installer automatique à prévoir</p> <p>Document de synthèse sur la réalisation du projet, comportant une explication sur la démarche de réalisation suivie, une explication précise sur le travail effectué par chacun, et une analyse critique et objective du projet.</p> <p>Tous les documents ci-dessus devront être postés sur la plate-forme de projets et imprimés en un exemplaire.</p> <p>REMARQUES :</p> <ul style="list-style-type: none"> <li>¥ TOUS LES DOSSIERS DOIVENT ETRE BONNE QUALITE ET REUNIS EN UN SEUL DOCUMENT POUR LE JURY EXTERIEUR</li> <li>¥ LES FICHIERS DOIVENT ETRE POSTES SUR LA PLATE-FORME DE PROJETS UN JOUR AVANT LA SOUTENANCE (MEME SI UN ACCES A UN COMPTE GITHUB OU BITBUCKET A ETE FOURNI)</li> <li>¥ L'APPLICATION JAVA DOIT ETRE LIVREE SOUS LA FORME D'UN FICHIER JAR AUTO-EXECUTABLE</li> </ul> <p>La soutenance aura normalement lieu le 22 juillet.</p>	<p><b>dimanche</b>  <b>20/07/2025</b>  <b>23h59</b></p>

5	Soutenance	
Durée de présentation par groupe :	<b>30 min</b>	Audience : <b>Publique</b>
Type de présentation :	<b>Démonstration</b>	
Précisions :	<b>Jurys extérieurs</b>	