

BD SERVEUR



*BOUZIDI Laura, CHAOUNI Mehdi, GAILLARD Cyril , HADI
Amina, ISSA Mahamat Abdelkerim, MARTINEAU Enzo,
ROGER Camille, VIVES Rémi, YAHYAOUI Rania*

SOMMAIRE

1. Objectif du groupe	3
1.1. Travail en commun des groupes “BD-INDEX” et “BD-SERVEUR”	3
1.2. Travail spécifique au groupe “BD-SERVEUR”	3
2. Organisation	5
2.1 Le calendrier prévisionnel	5
2.2 Le calendrier effectif	6
2.3 Le calendrier de tâches	7
2.4 Revues	8
2.5 Les outils de travail	11
2.6 La méthode utilisée	12
2.7 Contrôle qualité	13
3. Déroulement de la mission principale	14
4. Missions complémentaires	17
5. Difficultés rencontrées et solutions apportées	18
6. Conclusion	19

1. Objectif du groupe

1.1. Travail en commun des groupes “BD-INDEX” et “BD-SERVEUR”

Le premier objectif était la mise en place initiale de la base de données permettant de stocker le résultat du travail des groupes : “FILTRAGE DE TEXTE” (5), “ANALYSE SÉMANTIQUE” (6), de “PRÉDICTION/CLASSIFICATION” (7) et “ANALYSE DE TENDANCE/STATISTIQUES” (8). Une fois créée, notre mission a été de transmettre les informations nécessaires au groupe 8 et au groupe 9, ainsi que de répondre à tous les Use Cases minimaux présentés par le groupe Supervision en début de projet.

1.2. Travail spécifique au groupe “BD-SERVEUR”

L’objectif principal de notre groupe était de fournir les informations de la BD aux groupes “ANALYSE DE TENDANCE/STATISTIQUES” et “APPLICATION WEB”. Pour cela, nous avons dû écrire les requêtes et les procédures stockées répondant aux Use Cases ainsi qu’à la demande de ces deux groupes.

Nous avons donc recueilli au fur et à mesure de l'avancement du projet les besoins des groupes “ANALYSE DE TENDANCE/STATISTIQUES” et “APPLICATION WEB”. Par exemple :

- recueillir les mots-clés les plus courants de la semaine (Web)
- afficher le thème et le pourcentage du nombre d'articles de ce thème pour la semaine (Web)
- récupérer les données des vues pour que le groupe 8 puisse faire ces analyses (Stats)

Ensuite, une fois les besoins recueillis nous avons pu créer et mettre à jour les différentes requêtes et procédures afin de répondre à ces différents besoins.

```
--query 1 : View the most common keywords of the week

SELECT w.word, count(w.word)
FROM article a, word w, lemma l, position_word pw
WHERE w.id_lemma = l.id_lemma
AND w.id_word = pw.id_word
AND pw.id_article = a.id_article
AND a.date_publication BETWEEN CURRENT_DATE-7 AND CURRENT_DATE
GROUP BY w.word
ORDER BY 2 DESC LIMIT 5;
```

Exemple de requête SQL pour une demande dans la version 1 pour le groupe Web

Pour tester ces requêtes, nous avons dû créer un jeu de données afin de vérifier la cohérence des résultats obtenus par nos requêtes. Les requêtes ont donc été testées en local dans un premier temps, puis sur le serveur lorsque la base de données contenait un nombre de données suffisant pour exécuter les requêtes.

En parallèle à cette tâche, nous avons dû mettre en forme des fichiers Json afin de pouvoir communiquer avec les groupes Web et Statistique :

```
Json :
requête 3
{
  "1":{"feeling": "peur", "number": 10 },
  ...
  "7":{"feeling": "joie", "number":15}
}
```

Exemple de mise en forme du Json souhaité par le groupe Web

Après avoir créé les requêtes et mis en forme les fichiers Json, nous avons dû insérer les requêtes SQL dans le code python de l'API REST afin que les groupes "ANALYSE DE TENDANCE/STATISTIQUES" et "APPLICATION WEB" puissent récupérer les données souhaitées.

2. Organisation

2.1 Le calendrier prévisionnel

Janvier 2018	8	9	10	11	12		15	16	17	18	19
Nom des tâches											
Récupérer les informations des différents groupes											
Création de la base de données											
Création des Triggers											
Création des procédures											
Rédaction du rapport											
Maintenance et évolution de la BD											
Maintenance et évolution de l'API											

2.2 Le calendrier effectif

Janvier 2018	8	9	10	11	12		15	16	17	18	19
Nom des tâches											
Récupérer les informations des différents groupes											
Création des jeux de données test											
Création des procédures et requêtes groupe Web											
Création des procédures et requêtes groupe Stat											
Mise en forme des fichiers Json groupe Web											
Mise en forme des fichiers Json groupe Stat											
Insertion des requêtes MySQL dans l'API											
Mise à jour des procédures et requêtes groupe Web											
Mise à jour des procédures et requêtes groupe Stat											
Optimisation du code python groupe Stat											
Rédaction du rapport											
Maintenance et évolution de l'API											

2.3 Le calendrier de tâches

Initiales des membres du groupe	LB	MC	CG	AH	MI	EM	CR	RV	RY
Nom des tâches	Répartition des tâches								
Récupérer les informations des différents groupes									
Création des jeux de données test									
Recherches sur les vues matérialisées									
Création des procédures et requêtes groupe Web									
Création des procédures et requêtes groupe Stat									
Mise à jour des procédures et requêtes groupe Web									
Mise à jour des procédures et requêtes groupe Stat									
Mise en forme des fichiers Json groupe Web									
Mise en forme des fichiers Json groupe Stat									
Optimisation du code python groupe Stat									
Insertion des requêtes MySQL dans l'API Web									
Insertion des requêtes MySQL dans l'API Stat									
Rédaction du rapport									
Maintenance et évolution de l'API									

SQL API JSON Rapport BDD

2.4 Revues

Revue n°1 : 8 Janvier 2018

- Objectifs :
 - Installation des outils
 - Etablir les besoins des groupes
- Participants : Tous les membres du groupe
- Description :

Dans un premier temps, nous avons installé les logiciels nécessaires à la réalisation de nos différentes tâches prédéfinies dans le planning prévisionnel. Ensuite, nous avons pris contact avec les membres des autres groupes pour définir leurs besoins en données.

Revue n°2 : 9 Janvier 2018

- Objectifs :
 - Vues matérialisées
 - Requêtes statiques
 - API
- Participants : Tous les membres du groupe
- Description :

Nous avons divisé les membres en sous-groupes pour pouvoir répartir les différentes tâches. Une première tâche consistait à créer les requêtes SQL pour le groupe Web. Les données n'étant pas encore insérées dans la base de données, nous avons créé un jeu de test pour pouvoir tester les requêtes. Ensuite, nous avons fait des recherches sur les vues matérialisées afin de stocker et d'accéder plus facilement à certaines données. Enfin, nous avons continué notre débogage de l'api pour la communication entre le groupe web (g9) et le groupe statistique (g8).

Revue n°3 : 10 Janvier 2018

- Objectifs :
 - Requêtes dynamiques
 - Requêtes statiques
 - API
- Participants : Tous les membres du groupe
- Description :

Nous avons poursuivi l'écriture et les tests des requêtes pour le groupe Web. Nous avons, d'une autre part, commencé la mise en forme des fichiers pour le groupe statistique (g8). Enfin, nous avons réussi à établir une communication entre le groupe web et le groupe statistique. Pour faciliter l'intégration des requêtes et de l'appel des fonctions de l'api, nous nous sommes redirigés vers le microframework Flask.

Revue n°4 : 12 Janvier 2018

- Objectifs :
 - Requêtes dynamiques
 - Requêtes statiques
 - API
 - Rapport
- Participants : Tous les membres du groupe
- Description :

Nous avons terminé les tests de l'API pour la communication entre les différents groupes. Nous avons également débuté l'implémentation des différentes requêtes fonctionnelles dans l'API. Enfin, nous avons commencé à mettre en forme les fichiers de sorties des fonctions pour respecter les demandes du groupe Web. La rédaction du rapport a également démarré.

Revue n°5 : 16 Janvier 2018

- Objectifs :
 - Requêtes dynamiques
 - Requêtes statiques
 - API
 - Rapport
- Participants : Tous les membres du groupe
- Description :

Une partie du groupe a travaillé sur l'optimisation des requêtes créées ainsi que sur le respect de la charte de codage du groupe 10. Nous avons également pu tester quelques requêtes sur les données insérées dans la base de données. Le reste des membres ont continué à rédiger le rapport.

Revue n°6 : 18 Janvier 2018

- Objectifs :
 - Requêtes dynamiques
 - Requêtes statiques
 - API
 - Rapport
- Participants : Tous les membres du groupe
- Description :

Nous avons rajouté les dernières demandes de requêtes pour le groupe 9 et nous avons continué la maintenance de l'API. Nous avons également terminé la rédaction du rapport.

2.5 Les outils de travail

Les outils utilisés pendant la réalisation de notre mission sont :

- Trello pour la répartition des tâches, les tâches à faire, les tâches en cours de réalisation, les tâches terminées, et les revues ([Voir ici](#)).
- Github qui permet à chaque groupe d'avoir accès à tous les fichiers de tous les groupes du projet. Tous nos travaux y sont déposés régulièrement. Ainsi, on peut voir toutes les versions de travaux qui ont été réalisées.
- Slack qui nous sert de plateforme de communication entre étudiants, groupes et enseignants.
- Wampserver, environnement comprenant un serveur Apache, le système de gestion de base de données MySQL, un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL, ce qui nous a permis de tester nos requêtes en local.
- SourceTree qui offre une interface graphique et structurée des dépôts, permettant de voir l'historique des commits, et d'avoir la copie des travaux réalisés.
- Spyder pour le traitement des commandes python en local, ce qui permet d'effectuer les tests de nos fonctions et de respecter la charte de codage.

2.6 La méthode utilisée

Dans notre projet, nous avons utilisé la méthode AGILE. Tout a été spécifié dès le début du projet et on s'en tient à un cadre très rigide. Notre projet a été organisé en cycle de développement itératif et incrémental, auquel les groupes ont été intégrés et ont participé de façon active. L'objectif n'est alors plus de mener le projet à bien, mais de donner naissance à un produit fini, correspondant parfaitement au besoin.

Parmi les méthodes agiles existantes, on a utilisé la méthodologie Scrum qui est considérée comme un cadre ou « framework » de gestion de projet. Ce cadre est constitué d'une définition des rôles, de réunions et d'artefacts. La vie de notre projet est rythmée par un ensemble de réunions clairement définies et limitées dans le temps (timeboxing) :

- **Planification du Sprint** : c'est une réunion durant laquelle l'équipe sélectionne les éléments prioritaires à faire parmi la liste des tâches qu'elle pense pouvoir réaliser au cours de l'itération.

- **Revue de Sprint** : l'équipe fait une réunion à la fin de chaque sprint dont elle présente les fonctionnalités terminées, recueille les feedbacks du chef du projet et reçoit les instructions avant le prochain sprint.

- **Mêlée quotidienne** : il s'agit d'une réunion de synchronisation entre les chefs de projets de tous les groupes, qui se fait à la fin de chaque journée en 15 minutes, au cours de laquelle chacun répond principalement à 3 questions : « Qu'est ce que j'ai terminé depuis la dernière mêlée ? Qu'est ce que j'aurai terminé d'ici la prochaine mêlée ? Quels obstacles me retardent ? »

2.7 Contrôle qualité

Pour la conformité du projet, nous avons réalisé plusieurs test de qualité :

- Premièrement nous avons dû respecter la charte de codage réalisée par le groupe 10 (par exemple Pep8 pour les codes python).
- Deuxièmement nous avons dû effectuer des tests statiques et dynamiques sur nos codes pour vérifier qu'il n'y avait pas de code mort, et que toutes les fonctions retournaient le résultat désiré.
- Nous avons également créé des fonctions "type" que nous avons réutilisé dans plusieurs de nos codes de manière à rendre nos fichiers plus lisibles.
- Enfin, nous aurions dû envoyer nos codes au groupe 10 pour effectuer un contrôle qualité, cependant, cela n'a pas pu aboutir car nous étions "en bout de chaîne" ce qui a impliqué une modification des codes après la fin du délais imposé par le groupe qualité.

3. Déroulement de la mission principale

Dans la phase préliminaire, un calendrier prévisionnel, un schéma v0 de la base, un format de réponse au site ainsi qu'un format de requête à l'analyseur statistique sont mis en place. De plus, une première version de l'API a été conçue afin de permettre la communication entre notre groupe (3) et les groupes "ANALYSE DE TENDANCE/STATISTIQUES"(8), et "APPLICATION WEB" (9).

Ensuite, dans la phase de lancement, une réunion est organisée afin de comprendre les objectifs attendus. Le chef de notre groupe (Mehdi) répartit les tâches de chacun grâce à l'outil Trello. Les différentes missions remplies sont listées dans le calendrier des tâches (voir partie 2.3).

Les groupes "APPLICATION WEB" et "ANALYSE DE TENDANCE/STATISTIQUES" nous envoient des demandes de requêtes dont les résultats seront affichés sur le site.

Nos missions sont :

- Créer les requêtes qui permettent de collecter les données dans la base qui se trouve sur le serveur;

```
1 SELECT la.label, ((count(a.id_article) / (SELECT count(id_article)
2 FROM article))*100)
3 FROM article a, belong b, label la
4 WHERE b.id_label = la.id_label
5 AND b.id_article = a.id_article
6 AND a.date_publication BETWEEN CURRENT_DATE-7 AND CURRENT_DATE
7 ORDER BY 2 DESC LIMIT 1;
```

Exemple d'une requête pour le groupe WEB (9)

- Mettre en forme les requêtes pour répondre à la mise en page imposée par les deux groupes;


```

92 def tf_idf_label(name_label):
93     date_min_res = execute_query("""SELECT DATE_ADD(CURRENT_DATE,
94     INTERVAL -7 DAY) """)
95     date_max_res = execute_query("""SELECT CURRENT_DATE """)
96     date_max_res = str(date_max_res[0][0])
97     date_min_res = str(date_min_res[0][0])
98     query_word = """SELECT w.word
99     FROM word w, position_word pw, article a, belong b, label l
100    WHERE a.date_publication BETWEEN %s AND %s
101    AND a.id_article = pw.id_article
102    AND pw.id_word = w.id_word AND a.id_article = b.id_article
103    AND b.id_label = l.id_label AND l.label = %s
104    ORDER BY w.id_word;
105    """ % ("'" + date_min_res + "'", "'" + date_max_res + "'", "'"
106    + name_label + "'")
107    word_res = execute_query(query_word)
108    list_word = []
109    for i in range(len(word_res)):
110        list_word.append(str(word_res[i][0]))
111    res = {}
112    for word in word_res:
113        res[word] = tf_idf_word(word)
114    result = {}
115    result[name_label] = res
116    return res

```

Exemple d'une mise en page pour le groupe STAT (8)

```

1  {"period": "02/08/2017",
2   "Macron" : [[0.6],[0.3,0.4,0.5]],
3   "Tempete" : [[],[0.25,0.8,0.5]],
4   "Enfant" : [[0.5,0.6],[0.5,0.1,0.2]],
5   "Fleur" : [[0.9,0.6],[0.5]],
6   "Jardin" : [[0.3,0.6],[0.1,0.5]],
7   "Jouet" : [[0.5],[0.5,0.9,0.9]],
8   "Jeux" : [[0.6],[ ]],
9   "Magazine" : [[0.3,0.65],[0.1,0.5]],
10  "Noel" : [[0.5,0.6],[0.5]],
11  "Jour" : [[0.5],[0.1,0.2]],
12  "Test" : [[0.5,0.6],[0.5]]
13  }

```

Exemple de résultat attendu par le groupe ANALYSE DE TENDANCE/STATISTIQUES (8)

- Puis on transforme ces dictionnaires en format Json afin de les envoyer via l'API pour que les deux groupes puissent utiliser les données.
- Nous servons également de passerelle entre les deux groupes. Par exemple, le groupe "ANALYSE DE TENDANCE/STATISTIQUES" nous renvoie un fichier avec des calculs statistiques sur les données que nous transmettons au groupe "APPLICATION WEB".

Tout au long du projet, les tâches évoluent en fonction des besoins, ce qui a engendré le changement du calendrier prévisionnel vers le calendrier effectif qui décrit l'ajout, la modification ou la suppression de certaines tâches et le temps relatif à celles-ci. Les missions ont été répétées plusieurs fois, à cause des changements de base et des demandes.

4. Missions complémentaires

Durant toute la durée du projet, et notamment au début de celui-ci, nous avons dû travailler en collaboration étroite avec le groupe 2 : BD Index, qui gérait notamment la base de données. Cette collaboration a impliqué de l'aide des deux côtés pour mener à bien les missions des deux groupes : nous avons par exemple aidé à la formation du premier Modèle Conceptuel de Données (MCD), et à son développement tout au long du projet. La mise en place de vues matérialisées a nécessité notre aide pour déterminer ce qu'elles devaient contenir, leur utilisation dépendant principalement de nous.

5. Difficultés rencontrées et solutions apportées

- Parmi les difficultés rencontrées, notons tout d'abord les difficultés avec le langage informatique PHP pour implémenter l'API. Afin de diminuer le nombre de communication entre les groupes ainsi que pour une meilleure aisance avec le langage (tout cela en accord avec les enseignants), nous avons décidé d'utiliser le langage Python pour implémenter l'API.
- Un autre problème a été la mise en forme sous format json des fichiers répondant aux besoins du groupe 8 "ANALYSE DE TENDANCE/STATISTIQUES". Nous devions transmettre à ce groupe un nombre d'informations très important, conformément à leur demande. Cependant, cela donnait une complexité très importante à notre code Python, et des performances très faibles. La solution trouvée a été de spécifier correctement les demandes de chaque groupes de la "chaîne", pour à la fin de celle-ci, ne transmettre que les informations utiles, ce qui a diminué le temps d'exécution de nos fonctions de manière importante.
- Les modifications récurrentes du MCD et des tables par le groupe "BD-INDEX" étaient aussi une contrainte dans nos travaux. Une mise à jour permanente de nos requêtes était donc nécessaire.
- Un autre problème a été le manque de données dans la base (voire l'absence au début) qui ne nous permettait pas de tester toutes nos requêtes. Pour répondre à cette problématique, nous avons créé un petit jeu de données afin d'exécuter (et valider) nos requêtes.

6. Conclusion

Au cours de ce projet, nous devions parvenir à créer une API fonctionnelle pour la communication avec les autres groupes, ainsi que répondre à leurs besoins concernant les informations contenues dans la base de données. Nous avons réussi à remplir ces deux principaux objectifs durant le temps imparti.

Ce projet s'est révélé très enrichissant dans la mesure où il a consisté en une approche concrète des métiers auxquels nous nous préparons. En effet, la prise d'initiative, le respect des délais et le travail en équipe seront des aspects essentiels dans notre futur métier.

De plus, il nous a permis d'appliquer nos connaissances en base de données à un domaine pratique qu'est la création d'un site web.

Cependant, les contraintes de temps ne nous ont pas permis de parvenir à un résultat optimal. En effet, nos codes python restent assez complexe malgré les tentatives de simplifications, de même pour quelques requêtes qui restent relativement longues et laborieuses.