



# RAPPORT TECHNIQUE bdBOUM

2023

PRÉPARÉ PAR  
**LUIS DOUDEAU**  
**MAXYM CHARPENTIER**  
**LENNY DE NARDI**

---

*SAE 4.02 - Gestion de festival - Projet S4*  
**BDBOUM**

IUT ORLEANS

# **I - INTRODUCTION**

Ce rapport technique présente le développement de l'application de gestion d'inscription et d'organisation des différents participants pour le festival de bandes dessinées organise chaque année à Blois par l'association BD Boum. Actuellement, la saisie des données se fait sur le tableur Excel à partir de renseignements donnés sous forme de fiches à l'association par échange de mails ou courrier. L'association retourne ensuite une feuille de route complétée à chaque participant. Afin de simplifier et de moderniser ce processus, l'association souhaite informatiser la saisie et les traitements, ainsi que la feuille de route retournée à chacune des personnes. Ce rapport technique détaillera les différentes étapes du projet, et comment nous avons développé l'application, afin de répondre aux besoins de l'association BD Boum et de faciliter l'organisation de ce festival de bandes dessinées.

## **II - IDENTIFIER LES BESOINS**

Nous avons identifié une vingtaine de besoins clé que nous avons utilisé ensuite pour développer notre application, en voici la liste bien que non-exhaustive :

- Un système informatisé pour la saisie et le traitement des données.
- Un site web avec deux modules : SECRÉTAIRE/PARTICIPANT.
- Un moyen d'identification pour les participants pour accéder au site web.
- La saisie en ligne des informations par les participants.
- Un système gérable par les organisateurs pour gérer la saisie des informations.
- Des données utiles à la gestion du festival pour chaque profil (auteur, Invité, Exposant, Presse, Organisation, Bénévole).
- Les informations sur l'identité et les coordonnées de chaque participant.
- Les informations sur l'arrivée, le départ et les moyens de transport pour chaque participant.
- Les informations sur l'hôtel pour chaque participant.
- Les informations sur les repas pris pour chaque participant.
- Les créneaux de dédicaces pour les Auteurs.
- Les interventions pour chaque intervenant.
- Les horaires des événements tels que l'inauguration, les débats, les remises de prix, ...
- Les informations sur les navettes entre la gare et le festival.
- Les informations sur les voitures (plan d'accès, tickets de parking).
- Les repas disponibles et les créneaux horaires pour chaque repas.
- Les renseignements pour chaque profil (hôtel, restauration, créneaux de dédicaces).
- Le nombre total de repas pris globalement et dans chaque lieu pour chaque repas.
- Des room-lists pour chaque hôtel avec l'identité des occupants.
- Un système entièrement automatisé avec une bonne ergonomie et des traitements sûrs et vérifiables.

### **III - CADRER LE PROJET**

La création d'une application de telle ampleur nécessite d'adopter une très grande rigueur. En effet, l'organisation lors d'un projet informatique est très importante pour sa réussite. Pour répondre aux besoins identifiés, il est nécessaire de définir une méthodologie de travail rigoureuse, avec des étapes clairement définies pour la conception, le développement et le déploiement de l'application.

Il est également important de définir un plan de gestion de projet solide, avec des objectifs clairement définis, des échéances précises et des indicateurs de suivi pour évaluer la progression du projet. Une communication régulière et efficace entre les différents acteurs du projet est également essentielle pour garantir une collaboration harmonieuse et une gestion efficiente des éventuelles difficultés.

Nous avons commencé par définir les rôles de chacun dans ce projet : d'abord dédier le rôle de chef de projet. Ce rôle a été attribué à Luis DOUDEAU. De par son extrême motivation à réussir le projet, et son sens de l'organisation, il était le plus à même d'être nommé à ce rôle. Le rôle de scrum master (Rôle clé dans la méthodologie de développement agile Scrum. Le Scrum Master est chargé de faciliter l'équipe de développement dans l'application de la méthodologie Scrum et d'assurer que les pratiques et les règles de Scrum sont suivies de manière appropriée.) a été confié à Lenny DE NARDI. C'est par ses qualités en termes d'interaction humaine, communication, capacité à résoudre les conflits, qu'il était le plus à même d'être nommé pour ce rôle.

Nous avons ensuite créé divers moyens de partager le travail, de manière à travailler en collaboration à n'importe quel moment et n'importe quel lieu de travail.

Nous avons commencé par la création d'un dépôt Github privé réunissant les 3 développeurs de l'équipe, puis rajouté à ce dépôt les 4 professeurs référents que sont : Thomas PINSARD, Julien ARSOUZE, Gérard ROZSAVOLGYI et notre professeur principale du projet : Christophe LECHOPIER. Nous avons de plus créé un Dossier Google Drive afin d'y déposer tous les fichiers d'analyse (UML, Maquettes, fichier Gestion Projet, etc.).

## **IV - ANALYSE**

L'analyse lors d'un projet informatique est une étape cruciale pour la réussite du projet. Elle permet de bien comprendre les besoins et les attentes des utilisateurs et de proposer des solutions adaptées. L'analyse consiste à étudier en détail les différentes exigences, les contraintes, les fonctionnalités et les objectifs du projet.

Pour cela, plusieurs méthodes et outils sont utilisés comme les diagrammes de cas d'utilisation, les diagrammes de séquence, le MCD, les maquettes, les user stories, etc. L'analyse permet ainsi de définir le périmètre du projet, les fonctionnalités à développer et les spécifications techniques.

L'analyse a été menée en collaboration étroite avec notre professeur de projet afin de bien comprendre les besoins et les attentes. Cette collaboration permet également de valider les hypothèses et les choix techniques tout au long du projet et de ne pas faire fausse route.

Nous avons tout d'abord commencé par la base de notre application : La création de la base de données et donc d'un MCD. Nous avons utilisé le logiciel draw.io pour réaliser le MCD et les différents diagrammes nécessaires. Vous pourrez retrouver tous nos diagrammes d'analyse dans le répertoire : `./GESTION_DE_PROJET/` . Notre équipe de développement a aussi été à la charge de créer un cahier des charges de l'application, il se trouve dans le répertoire `./GESTION_DE_PROJET/ORG/` .

Nous avons aussi créé des maquettes de développement, qui n'ont à l'heure actuelle plus de ressemblance avec l'application, mais qui nous ont tout de même permis de se poser les bonnes questions notamment sur l'organisation des pages et les choix techniques et esthétiques. Nous avons créé d'autre type de diagramme comme des diagramme d'activité, de séquences ou encore des DOM.

Nous avons aussi créé une matrice risque, une matrice RACI ainsi qu'un GANTT afin de prévoir les risques qui menacent le projet, la vision que doit avoir chaque développeur sur toutes les étapes du projet ainsi que l'organisation du projet semaine par semaine et permettre de cadrer le projet dans le temps. Tous ces fichiers sont disponibles comme dit précédemment dans le répertoire `./GESTION_DE_PROJET/` .

## **IV - DÉVELOPPEMENT**

Nous avons travaillé selon un processus rigoureux pour la création de notre application en Flask. Tout d'abord, après avoir identifié les besoins et analysé les exigences, nous avons pu passer à l'étape du développement. Nous avons commencé par créer une architecture de fichier claire et bien organisée, spécifique au développement d'application Flask, avec des dossiers dédiés contenant des types de fichiers spécifiques. Par exemple, le dossier 'static' contient tous les scripts JS, les images de l'application ainsi que les fichiers CSS, tandis que le dossier 'template' contient toutes les pages HTML de l'application.

Nous avons ensuite développé le script SQL pour créer la base de données. Dès le début, nous avons créé un jeu de test qui nous permettra tout au long du projet d'avoir une base de données remplie qui servira d'outil de test. Nous avons créé le fichier 'App.py', qui est le fichier de création de l'application Flask. C'est dans ce fichier que nous avons créé l'application avec cette ligne de code : `app = Flask(name)`.

Dans le même temps, nous avons créé le fichier 'ConnexionPythonSQL.py', qui, grâce à l'import de l'ORM SQLAlchemy, relie notre base de données à notre application. Cette étape est plutôt simple mais très importante car nous allons sans cesse communiquer avec la base de données. Il est donc nécessaire que la liaison soit entièrement fonctionnelle. Ce fichier contient à la fois la connexion à la base de données et toutes les requêtes dont nous avons besoin pour afficher les données dans notre application.

Nous avons ensuite créé tous les objets Python nécessaires. En effet, pour utiliser notre ORM, nous avons besoin de créer des classes (des objets par la suite) qui possèdent les mêmes propriétés que les entités de la base de données. Par exemple, nous devons créer une classe 'SECRETAIRE' avec comme attribut de classe : `idP`, `prenomP`, `nomP`, `emailP`, `mdpP`. Nous avons donc créé autant de fichiers Python que d'entités dans notre MCD. À noter que les types de la base de données doivent être respectés lors du "mapping" : une `idP` dans le MCD qui est de type INT, doit être un INTEGER dans notre classe. Une fois le processus de mapping fonctionnel, nous avons pu commencer à créer des requêtes et tester leur bon fonctionnement.

Le plus dur est souvent dans les imports de fichier, car une petite erreur entraîne un non-fonctionnement de l'application. Mais nous n'avons pas rencontré de gros problèmes. Nous avons utilisé dans ce projet le système d'héritage qui possède plusieurs avantages : le polymorphisme, l'encapsulation et une maintenance facilitée.

Nous avons utilisé ce concept d'héritage, car il existe plusieurs types de rôles dans ce festival. Le cas le plus parlant est celui du PARTICIPANT et SECRETAIRE. Ce sont deux rôles différents, mais qui possèdent tout de même des informations similaires. Nous avons donc décidé de créer une entité supplémentaire nommée UTILISATEUR qui est la classe/entité mère de deux classes : SECRETAIRE et PARTICIPANT. Vous pourrez constater sur notre MCD que d'autres héritages existent par rapport aux types de rôle qui existent.

En ce qui concerne la création des différentes pages (Participants, Navette, Mangeur, ...) du module Secrétaire, Maxym CHARPENTIER et Lenny DE NARDI ont principalement travaillé dessus. Quant à Luis DOUDEAU, il s'est plutôt concentré sur le module Participant en créant des formulaires, en mettant en base de données les informations, en générant la feuille de route, etc.

Pour obtenir plus d'informations sur le développement d'une fonctionnalité, d'une page ou sur le travail de chaque développeur, vous pouvez consulter le fichier [Carnet\\_de\\_bord.pdf](#) qui se trouve dans le dossier /GESTION\_DE\_PROJET/ORG/. Si vous avez d'autres questions auxquelles nous n'avons pas répondu, n'hésitez pas à nous contacter via nos adresses universitaires.

## **VI - EXPÉRIENCE UTILISATEUR**

Lorsque vous développez une application destinée à des utilisateurs peu familiers avec les outils numériques, il est crucial d'assurer une expérience utilisateur fluide et intuitive. Dans le cas contraire, les utilisateurs pourraient faire des erreurs et adopter des comportements inappropriés. Pour éviter cela, vous pouvez incorporer des éléments dans l'application qui minimisent la confusion et empêchent les utilisateurs d'avoir l'impression de faire des erreurs à chaque interaction. En clair, l'objectif est de rendre l'utilisation de l'application aussi simple et agréable que possible pour l'utilisateur.

Voici une liste des éléments clés de notre application qui contribuent à offrir une expérience utilisateur optimale :

- L'animation de zoom qui se produit lors du passage de la souris sur un élément interactif (bouton, élément cliquable, etc.) et qui peut être réalisée en utilisant la propriété "transform: scale" en CSS.
- Les couleurs des éléments sont soigneusement choisies pour être bien distinguées du fond, agréables et cohérentes avec le reste de l'application.
- Les logos (images) sur certains éléments aident à expliquer l'intitulé, l'objectif et l'utilité de la page, comme sur la page d'accueil pour la secrétaire.
- Les boutons sont de différentes couleurs pour prévenir l'utilisateur de la conséquence de leur clic, par exemple un bouton rouge pour supprimer, retour en arrière, et vert ou bleu pour valider, continuer, ajouter, etc.
- La recherche automatique est effectuée lors de la saisie des aéroports et gares dans la page de saisie des transports utilisés par le participant, pour s'assurer que le nom de l'aéroport et/ou de la gare est correct.
- Les cases de repas (repas où le participant sera présent) sont automatiquement cochées, ce qui permet au participant de vérifier, potentiellement modifier, puis de valider.
- Des éléments "Plus d'information" sont inclus pour aider l'utilisateur à comprendre la saisie qu'il doit effectuer.
- Les informations de l'utilisateur sont mises en cache et chargées automatiquement lorsque l'utilisateur arrive sur une nouvelle page. Par exemple, un retour en arrière ou un retour sur l'application quelques jours après avoir rempli le formulaire.
- Lorsqu'une ligne est sélectionnée dans un tableau dynamique (DataTable JS), la couleur de fond de la ligne change pour mieux signaler qu'elle a été sélectionnée.
- Dans la page "Invité" du module Secrétaire, un bouton "Sélectionner tous" est présent pour faciliter la sélection de tous les participants en cas de grand nombre, sans avoir à les cocher individuellement.
- Le bouton "Menu" est présent dans le header de l'application et permet à la secrétaire de retourner directement à la page d'accueil sans avoir à utiliser les flèches de retour arrière du navigateur.

## **VII - CONCLUSION**

Le projet de gestion de festival pour l'association BD Boum a été une expérience riche en enseignements en matière de développement d'application web. Il a permis de mettre en pratique les connaissances acquises sur des technologies telles que Python et le framework Flask, JS et Ajax, ainsi que sur les bases d'une architecture d'application web (BDD, utilisation d'une ORM, ...). Notre travail a répondu aux besoins, à savoir la mise en place d'une application web qui permet d'optimiser la saisie et le traitement des données et l'automatisation d'un maximum de tâches.

Le choix de la technologie Flask pour la mise en place de l'application web a été judicieux, car elle a permis de mettre en place un développement rapide et efficace de l'application. L'utilisation de JS a été bénéfique pour la création d'une interface utilisateur interactive et dynamique, permettant de faciliter la saisie des données par les participants et les organisateurs.

Le projet a également permis de se familiariser avec les exigences et les contraintes de la gestion d'un événement de grande envergure, en prenant en compte les différents profils de participants et leurs besoins spécifiques en termes de transport, hébergement, repas et dédicaces. La création d'une interface utilisateur ergonomique et intuitive a été essentielle pour faciliter l'expérience utilisateur.

En somme, la mise en place de cette application web de gestion de festival a permis de mettre en pratique des compétences clés en matière de développement d'application web, tout en répondant aux besoins spécifiques d'une association organisatrice d'un événement de grande envergure. Cette expérience a été bénéfique pour acquérir de nouvelles compétences et pour renforcer les connaissances existantes, tout en fournissant une solution efficace et automatisée pour la gestion de ce type d'événement.