

Rapport Projet : Architecture des composants d'entreprise

2023/2024

**Réalisation d'une application de réservation
des billets des évènements**

Ingénierie Informatique et Réseaux

Réalisé par :

ALAOUI Mohamed

ARRAJ Mohamed

LAMDIBIH Ilhame

Encadré par :

Mr. LACHGAR Mohamed

Table de Matières

Table de Matières	2
Liste des figures	4
Introduction	1
CHAPITRE 1 :	2
CONTEXTE GENERALE DU PROJET.....	2
1 Aperçus du projet :.....	3
1.1 Introduction :	3
1.2 Problématiques :	3
1.3 Solution proposée :	3
1.4 Cycle de vie du projet :.....	3
2 Importance de l'architecture microservices.....	4
3 Conclusion :	5
CHAPITRE 2 :	6
ARCHITECTURE ET CONCEPTION	6
1 Introduction.....	7
2 Architecture Microservices	7
2.1 Microservice d'Authentification et de paiement en Django	7
2.2 Microservice de Gestion des Événements en Spring Boot.....	8
2.3 Microservice de Gestion des Réservations	8
2.4 Conclusion	8
CHAPITRE 4 :	10
IMPLEMENTATION ET MISE EN OEUVRE	10
1 Conteneurisation avec DOCKER	11
1.1 Avantage de Docker	11
1.2 Implémentation de Docker :	12
2 CI/CD avec Jenkins :	12

3	Déploiement automatique NGROK:	13
4	Intégration de SonarQube :	14
5	Les principales interfaces :	15
5.1	Interface d'accueil :	15
5.2	Interface Connexion et d'Inscription :	16
5.3	Interface des listes des évènements :	17
5.4	Interface Détail évènement :	18
5.5	Interface Cart :	19
5.6	Interface Paiement :	20
5.7	Interface liste des billets :	21
	Conclusion.....	22

Liste des figures

Figure 1: Architecture projet	9
Figure 2 : Implémentation Docker	12
Figure 3 :Implémentation Jenkins	12
Figure 4 : Implémentation NGROK.....	13
Figure 5 : Implémentation validé	13
Figure 6 : Après la commande push.....	14
Figure 7: Statistique de l'interface sonar Scanner	14
Figure 8: Interface Accueil.....	15
Figure 9 : Interface Login.....	16
Figure 10 :Interface d'inscription	16
Figure 11 :Liste des Evènement	17
Figure 12 : Détail événement	18
Figure 13 : Interface Cart	19
Figure 14 :Interface de paiement.....	20
Figure 15 : Interface liste des billets	21

Introduction

À l'ère contemporaine marquée par l'ubiquité du numérique, la recherche de moyens innovants pour faciliter et améliorer notre expérience lors d'événements culturels, sportifs, et de divertissement devient essentielle. L'évolution rapide des technologies de l'information et de la communication offre une opportunité unique de repenser la manière dont nous réservons nos places pour des expériences variées telles que des projections cinématographiques, des matches de football, des concerts, et bien d'autres.

Dans ce contexte, notre projet intitulé "Réalisation et conception d'une plateforme de réservation de billets pour des événements divers" se positionne comme une réponse novatrice aux défis contemporains liés à la réservation d'événements. L'objectif central de cette initiative est de créer une plateforme robuste, conviviale et complète, destinée à simplifier le processus de réservation tout en offrant une expérience utilisateur optimale.

L'évolution rapide des préférences des consommateurs et la diversification croissante des événements exigent une approche proactive. Notre plateforme ambitionne de répondre à ces besoins en proposant une solution unifiée, permettant aux utilisateurs de réserver facilement des billets pour des événements divers, que ce soit une sortie au cinéma, la participation à un match de football, ou la réservation de places pour un concert. En alignement avec les tendances actuelles, notre plateforme vise à offrir une expérience intuitive, personnalisée et sécurisée.

CHAPITRE 1 :

CONTEXTE GENERALE DU PROJET

1 Aperçus du projet :

1.1 Introduction :

À l'ère contemporaine où la technologie façonne notre quotidien, notre projet aspire à révolutionner la manière dont les individus réservent leurs places pour des événements variés tels que des projections cinématographiques, des matches de football, des concerts, et bien d'autres. À travers la création d'une plateforme de réservation de billets innovante et complète, nous cherchons à simplifier et à améliorer l'expérience de participation à ces événements, tout en offrant une solution transparente et adaptée aux besoins de notre public cible.

1.2 Problématiques :

Le cœur de notre projet réside dans la conception d'une plateforme de réservation qui répond efficacement aux diverses exigences des utilisateurs en matière de gestion de leurs participations à des événements. Nous sommes confrontés au défi de développer une solution intégrée, harmonisant les fonctionnalités de réservation, de gestion des places, et de personnalisation de l'expérience, tout en assurant une interface simple et intuitive. Notre plateforme doit être suffisamment flexible pour s'adapter aux besoins de différents utilisateurs, des amateurs occasionnels aux passionnés assidus.

1.3 Solution proposée :

Pour relever ces défis, notre solution repose sur le développement d'une plateforme web complète et intuitive. Cette plateforme permettra aux utilisateurs de créer des profils personnalisés, facilitant ainsi la réservation et la gestion de leurs places pour différents types d'événements. Elle mettra à disposition une interface conviviale pour la sélection des sièges, des informations détaillées sur les événements.

1.4 Cycle de vie du projet :

La gestion du cycle de vie de notre plateforme de réservation est cruciale pour garantir sa qualité tout en maîtrisant les coûts du projet. Nous avons choisi un modèle en cascade, structuré en phases distinctes :

- **Analyse et Spécification** : Cette étape initiale consiste à définir les besoins et les fonctionnalités de la plateforme en étudiant attentivement les exigences des utilisateurs.
- **Conception** : La phase de conception implique la modélisation de l'architecture de la plateforme, utilisant des outils tels que les diagrammes d'interaction et d'interface utilisateur.
- **Implémentation** : Une fois la conception établie, nous passerons à l'implémentation, développant le code source et intégrant les fonctionnalités essentielles.
- **Tests et Validation** : Les tests seront effectués à chaque étape pour garantir le bon fonctionnement de la plateforme, de la réservation des billets à l'interface utilisateur, en passant par la communication avec les organisateurs.

2 Importance de l'architecture microservices

Ce projet vise à révolutionner la manière dont les individus participent à des événements, en offrant une plateforme de réservation moderne et efficace, adaptée aux besoins variés de notre société dynamique.

L'importance de l'architecture microservices dans notre projet de conception et réalisation d'une plateforme de réservation de billets pour divers événements est cruciale pour assurer la flexibilité, la scalabilité et la maintenabilité de notre solution. L'architecture microservices adoptée constitue le fondement structurel de notre plateforme, offrant des avantages significatifs dans la gestion des différentes fonctionnalités et services qu'elle propose.

En optant pour une architecture microservices, nous avons découpé notre application en modules autonomes, indépendants et spécialisés, chacun étant responsable d'une tâche spécifique. Cette approche permet une plus grande agilité dans le développement, la mise à jour et la maintenance de la plateforme. Chaque microservice peut être développé, déployé et évolué de manière indépendante, facilitant ainsi l'adaptation rapide aux changements et l'ajout de nouvelles fonctionnalités.

La scalabilité horizontale est également un avantage majeur de l'architecture microservices. En permettant le déploiement indépendant de chaque service, notre plateforme peut facilement s'adapter à des charges variables, assurant une expérience utilisateur fluide même lors de pics

d'utilisation. Cette approche offre une souplesse précieuse pour répondre à la croissance de la demande et garantir la performance de la plateforme.

En outre, l'architecture microservices favorise une maintenance simplifiée. En cas de modification ou de correction nécessaire dans un microservice spécifique, l'impact sur le reste de la plateforme est minimisé. Cela permet des mises à jour plus rapides et moins risquées, assurant ainsi la disponibilité continue de notre plateforme.

En résumé, l'adoption de l'architecture microservices dans notre projet joue un rôle crucial en assurant la flexibilité, la scalabilité et la maintenabilité de la plateforme de réservation. Elle offre une base solide pour répondre efficacement aux défis liés à la diversité des événements, aux fluctuations de la demande et aux évolutions futures de notre projet.

3 Conclusion :

Dans ce chapitre on a présenté la problématique, les lacunes qu'il comprend ainsi que les solutions que nous proposons pour pallier ces problèmes.

Beaucoup des leçons tirées d'auprès cette étude, il reste maintenant d'étudier les Méthodes et bien spécifier les besoins pour la réalisation d'une nouvelle solution plus efficace et pertinente.

CHAPITRE 2 :

ARCHITECTURE ET CONCEPTION

1 Introduction

L'élaboration d'une architecture robuste et la conception réfléchie des différents composants de notre système sont essentielles pour garantir la performance, la flexibilité et la facilité de maintenance de notre solution.

Ce chapitre explorera les choix d'architecture, en mettant particulièrement l'accent sur l'adoption de l'architecture microservices, et examinera comment cette approche contribue à répondre aux défis spécifiques liés à la réservation d'événements. Nous plongerons également dans la conception détaillée de notre plateforme, en analysant les décisions prises pour chaque aspect, des fonctionnalités de réservation à la gestion des utilisateurs, afin de créer une expérience utilisateur cohérente et intuitive.

2 Architecture Microservices

L'architecture microservices adoptée pour notre plateforme de réservation d'événements repose sur une approche modulaire et évolutive, permettant une gestion optimale des différentes fonctionnalités du système. Cette structure fragmente l'application en plusieurs microservices autonomes, chacun se concentrant sur une tâche spécifique, favorisant ainsi une maintenance simplifiée, une évolutivité accrue et une résilience face aux changements.

2.1 Microservice d'Authentification et de paiement en Django

Le premier microservice se consacre à l'authentification des utilisateurs et aussi au paiement des billets, implémenté avec le framework Django. Assurant la gestion sécurisée des comptes utilisateur, ce microservice offre une expérience d'authentification et d'autorisation transparente. En centralisant cette fonctionnalité, nous garantissons la cohérence et la sécurité dans l'accès à l'ensemble de la plateforme. Il aussi prend en charge la gestion des transactions financières liées aux réservations. Intégrant des fonctionnalités de paiement sécurisées, ce microservice assure une expérience de paiement fluide et transparente pour les utilisateurs, tout en facilitant l'intégration avec différents fournisseurs de services de paiement.

2.2 Microservice de Gestion des Événements en Spring Boot

Le deuxième microservice, construit avec Spring Boot, est dédié à la gestion complète des événements proposés sur la plateforme. De la création à la mise à jour en passant par la gestion des disponibilités, ce microservice offre une architecture modulaire, permettant une adaptation rapide aux changements dans le catalogue d'événements.

2.3 Microservice de Gestion des Réservations

Le troisième microservice, basé sur Spring Boot, se concentre sur la gestion des réservations, assurant la coordination efficace entre les utilisateurs et les événements réservés. Il gère les transactions de réservation, les confirmations et les annulations, optimisant ainsi la disponibilité des places et offrant une expérience utilisateur cohérente.

2.4 Conclusion

L'utilisation de microservices dans notre architecture garantit une extensibilité et une flexibilité considérables. Chaque microservice opère de manière indépendante, facilitant le déploiement, la mise à jour et la maintenance, tout en permettant une évolution continue de la plateforme en réponse aux besoins changeants de nos utilisateurs. Cette approche modulaire contribue à la robustesse et à la scalabilité de notre solution de réservation d'événements.

4. Mécanismes de Communication entre les Microservices :

Dans notre architecture microservices, la communication entre les différentes composantes s'articule autour de mécanismes efficaces, dont le RESTTemplate occupe une place centrale pour faciliter l'échange de données de manière synchrone. Ci-dessous, nous détaillerons les principaux mécanismes utilisés dans notre système.

4.1 API RESTful avec RESTTemplate :

L'API RESTful demeure le fondement de notre communication inter-microservices, avec le RESTTemplate jouant un rôle clé. Intégré à notre architecture, le RESTTemplate simplifie les

requêtes HTTP entre les microservices. Chaque microservice expose des endpoints REST standardisés, et le RESTTemplate permet d'effectuer des appels HTTP pour récupérer des données, les modifier ou en envoyer de manière efficace.

4.2 Protocole HTTP/HTTPS :

La communication entre nos microservices s'opère à travers le protocole HTTP/HTTPS, assurant une transmission sécurisée des données. Ce choix de protocole standard facilite les interactions entre les différentes composantes tout en préservant la simplicité et la fiabilité de la communication synchrone.

4.3 Format de Données JSON :

Les données échangées entre les microservices sont structurées en utilisant le format JSON (JavaScript Object Notation). Ce format léger et lisible assure une interchangeabilité fluide des données, tout en facilitant la manipulation des informations entre les différents composants du système.

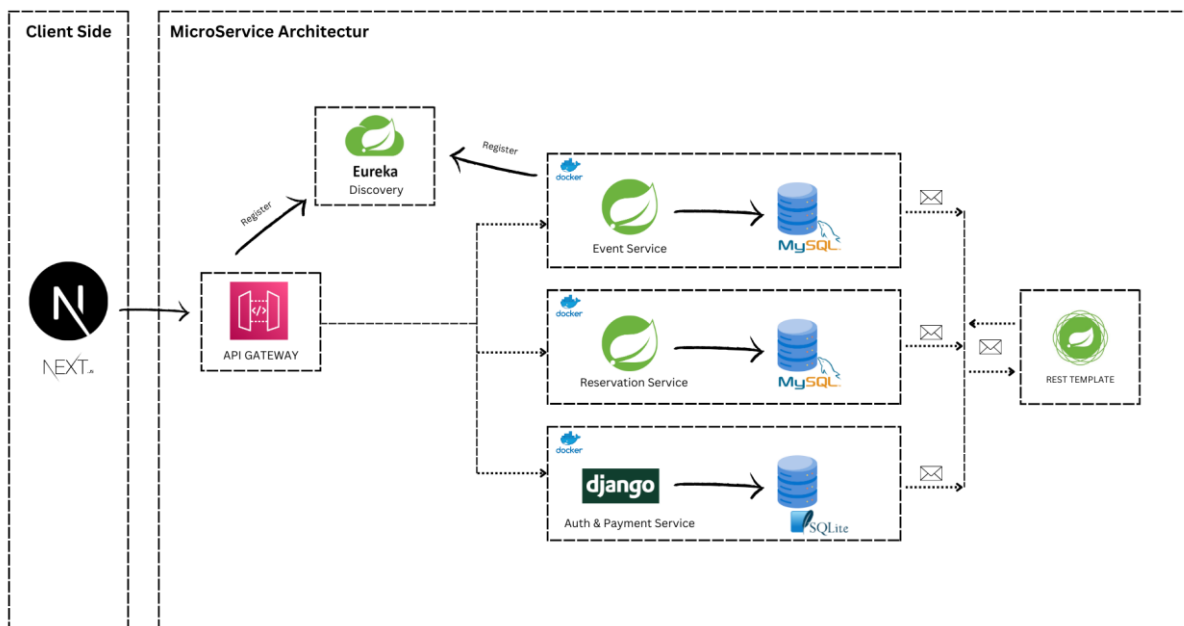


Figure 1: Architecture projet

CHAPITRE 4 :

IMPLEMENTATION ET MISE EN OEUVRE

1 Conteneurisation avec DOCKER

1.1 Avantage de Docker

Docker offre plusieurs avantages significatifs, ce qui explique en grande partie sa popularité croissante dans le domaine du développement logiciel et de l'administration système. Voici quelques-uns des principaux avantages de Docker :

Isolation des Environnements : Docker utilise des conteneurs légers et isolés qui encapsulent une application et ses dépendances. Cette isolation permet d'éviter les conflits entre les différentes applications et garantit une exécution cohérente, quel que soit l'environnement.

Portabilité : Les conteneurs Docker sont portables, ce qui signifie qu'ils peuvent être exécutés de manière cohérente sur n'importe quel environnement prenant en charge Docker, qu'il s'agisse d'un ordinateur local, d'un serveur en nuage ou d'un centre de données. Cela facilite le déploiement et la gestion d'applications sur diverses plateformes.

Rapidité de Déploiement : Les conteneurs Docker peuvent être créés rapidement, démarrés et arrêtés en quelques secondes. Cela permet une mise en production plus rapide, des déploiements plus flexibles et une utilisation plus efficace des ressources.

Gestion des Dépendances : Docker permet de spécifier et d'isoler les dépendances logicielles requises par une application dans son propre conteneur. Cela élimine les problèmes liés aux conflits de version et facilite la gestion des dépendances au niveau de l'application.

Évolutivité Horizontale : Les conteneurs Docker peuvent être orchestrés et gérés à l'échelle à l'aide d'outils tels que Docker Swarm ou Kubernetes. Cela permet une évolutivité horizontale facile, où de nouveaux conteneurs peuvent être ajoutés ou retirés en fonction des besoins de charge.

Gestion des Versions : Docker propose un système de gestion des images, ce qui signifie que les différentes versions d'une application peuvent être encapsulées dans des images distinctes. Cela facilite le suivi des versions et la gestion des mises à jour sans impact sur d'autres parties du système.

Facilité de Collaboration : Grâce à la portabilité et à la gestion des versions, Docker facilite la collaboration entre les équipes de développement et les équipes opérationnelles. Les

développeurs peuvent partager des images Docker avec des configurations prédéfinies, garantissant une cohérence dans les différents environnements.

Économie de Ressources : Les conteneurs Docker partagent le noyau du système d'exploitation de l'hôte, ce qui les rend plus légers par rapport aux machines virtuelles. Cela permet une utilisation plus efficace des ressources système et une meilleure densité d'applications sur un même serveur.

1.2 Implémentation de Docker :

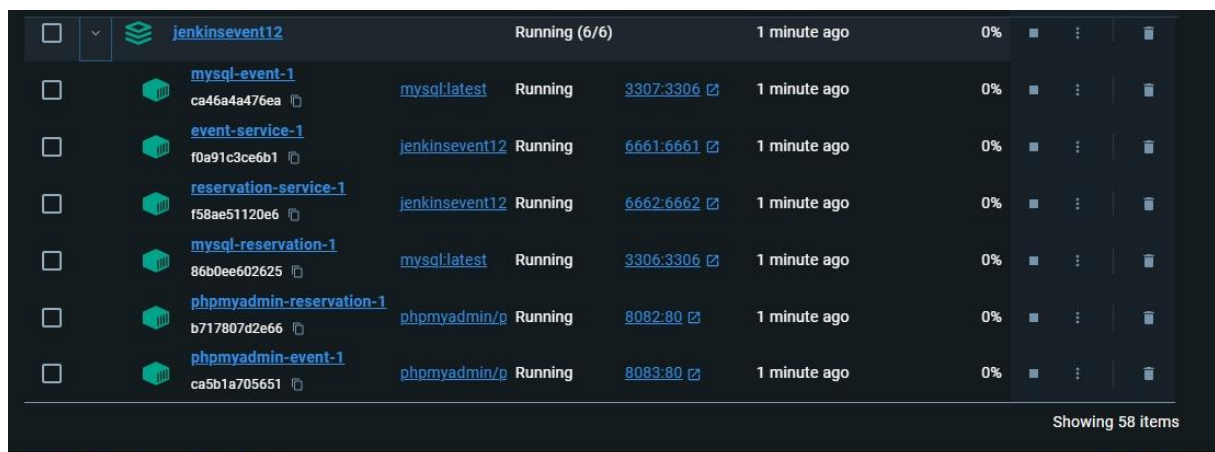


Figure 2 : Implémentation Docker

2 CI/CD avec Jenkins :

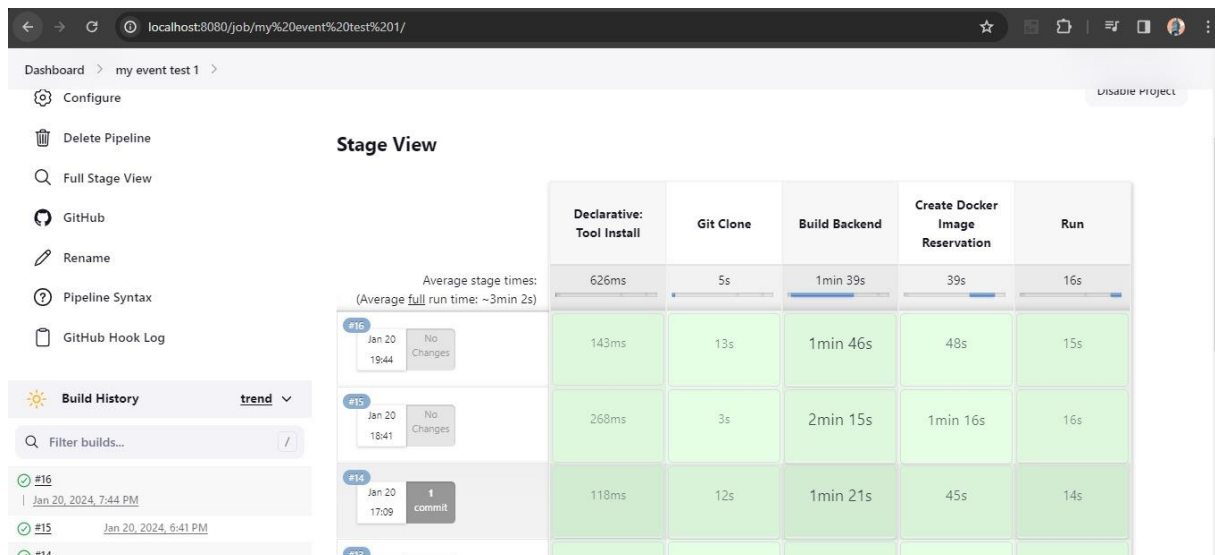


Figure 3 : Implémentation Jenkins

3 Déploiement automatique NGROK:

```
ngrok (Ctrl+C to quit)
Build better APIs with ngrok. Early access: ngrok.com/early-access

Session Status      online
Account             ilhame (Plan: Free)
Version             3.5.0
Region              Europe (eu)
Latency              83ms
Web Interface       http://127.0.0.1:4040
Forwarding           https://c829-105-69-49-121.ngrok-free.app -> http://localhost:8080

Connections          ttl    opn    rt1    rt5    p50    p90
                   97     0      0.03   0.13   0.13   0.61

HTTP Requests
-----
POST /github-webhook/                200 OK
POST /widget/BuildQueueWidget/ajax   200 OK
POST /widget/ExecutorsWidget/ajax    200 OK
GET /                                200 OK
GET /i18n/resourceBundle              200 OK
POST /manage/configSubmit              302 Found
POST /manage/descriptorByName/hudson.plugins.gradle.injection.InjectionConfig/checkServer 200 OK
POST /manage/descriptorByName/hudson.plugins.gradle.enriched.EnrichedSummaryConfig/checkHttpClientTimeoutInSeconds 200 OK
GET /adjuncts/e2681170/lib/credentials/select/select.js 200 OK
GET /i18n/resourceBundle              200 OK
```

Figure 4 : Implémentation NGROK

GitHub Pull Requests

Published Jenkins URL

☒ Actualise local repo on factory creation

Save

Apply

Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

✓ <https://c829-105-69-49-121.ngrok-f...> (push)

Edit

Delete

Figure 5 : Implémentation validé

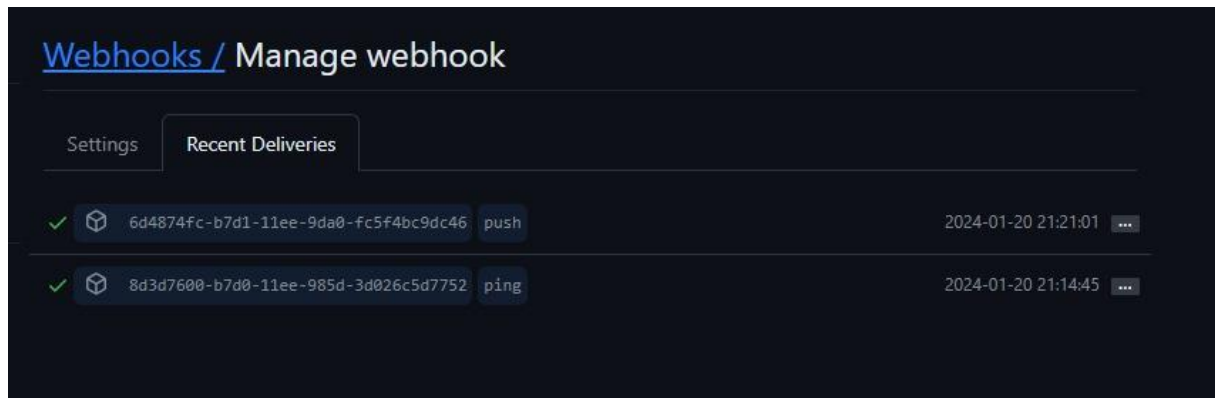


Figure 6 : Après la commande push

4 Intégration de SonarQube :

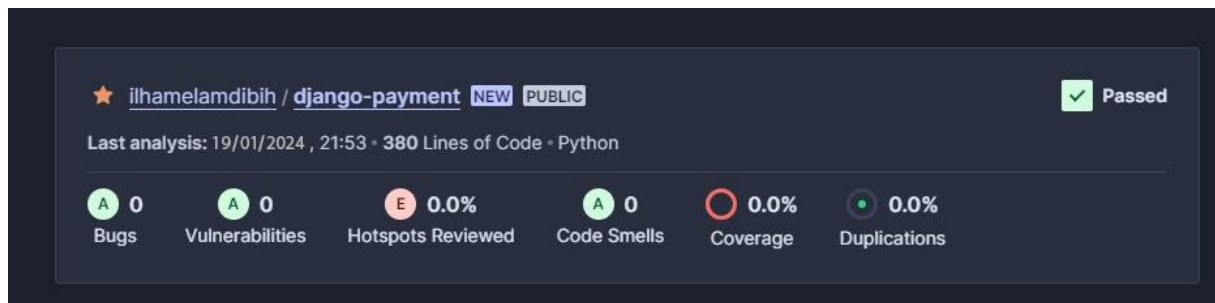


Figure 7: Statistique de l'interface sonar

5 Les principales interfaces :

5.1 Interface d'accueil :

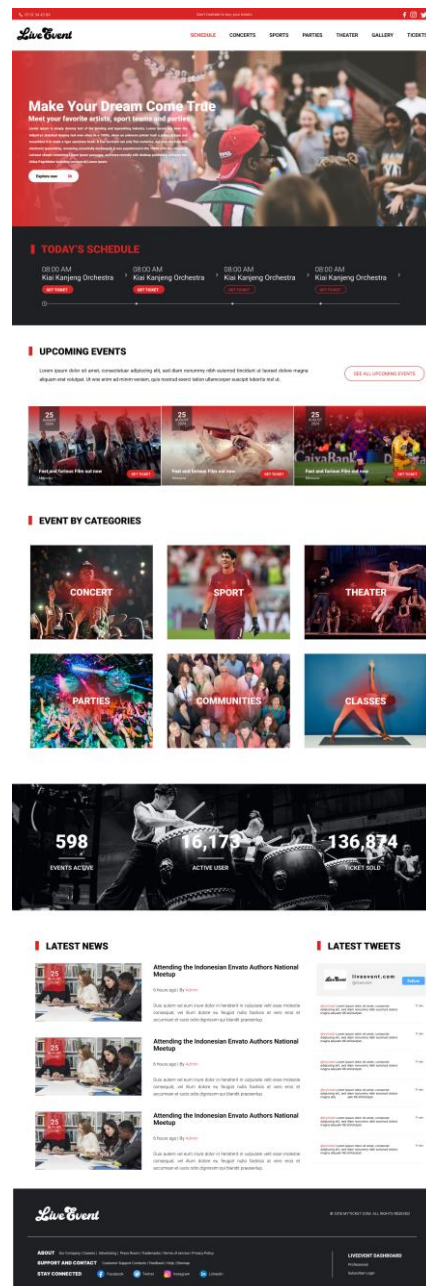


Figure 8: Interface Accueil

Cette interface permet à l'utilisateur de consulter les différents événements, la disponibilité des billets, les horaires des événements et les catégories qui existent.

5.2 Interface Connexion et d'Inscription :

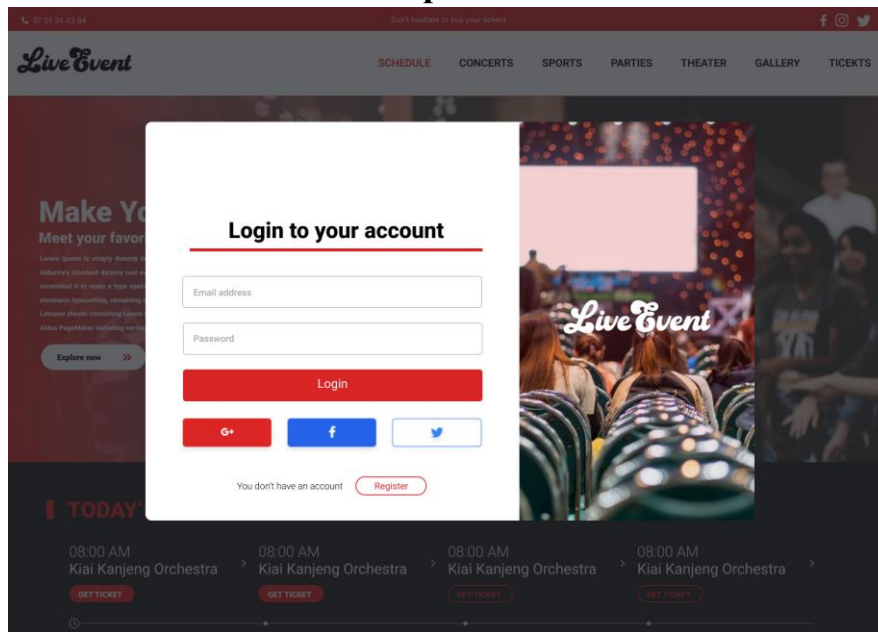


Figure 9 : Interface Login

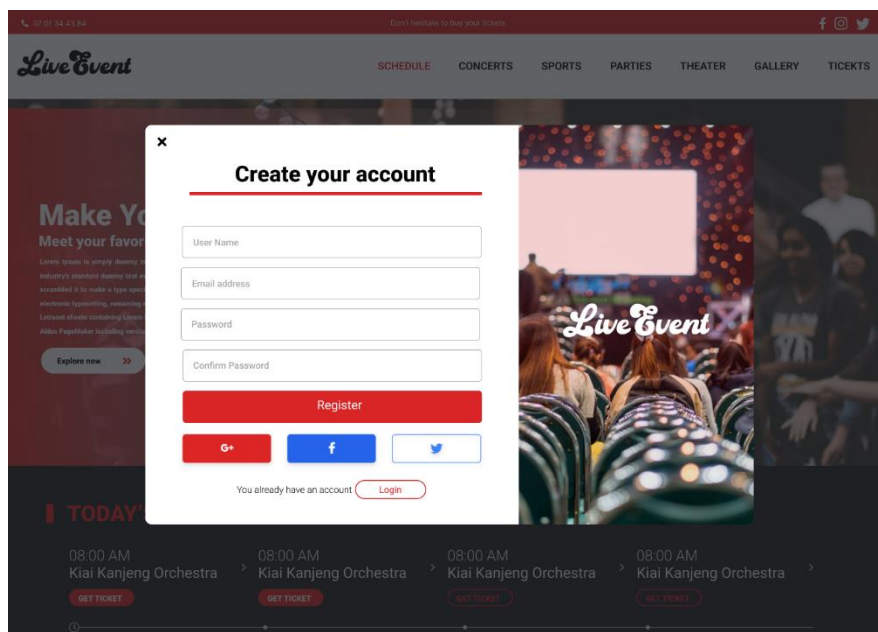


Figure 10 :Interface d'inscription

Cette interface permet l'authentification de l'utilisateur sur la plateforme afin qu'il puisse manipuler les événements, les réservations et les billets

5.3 Interface des listes des évènements :

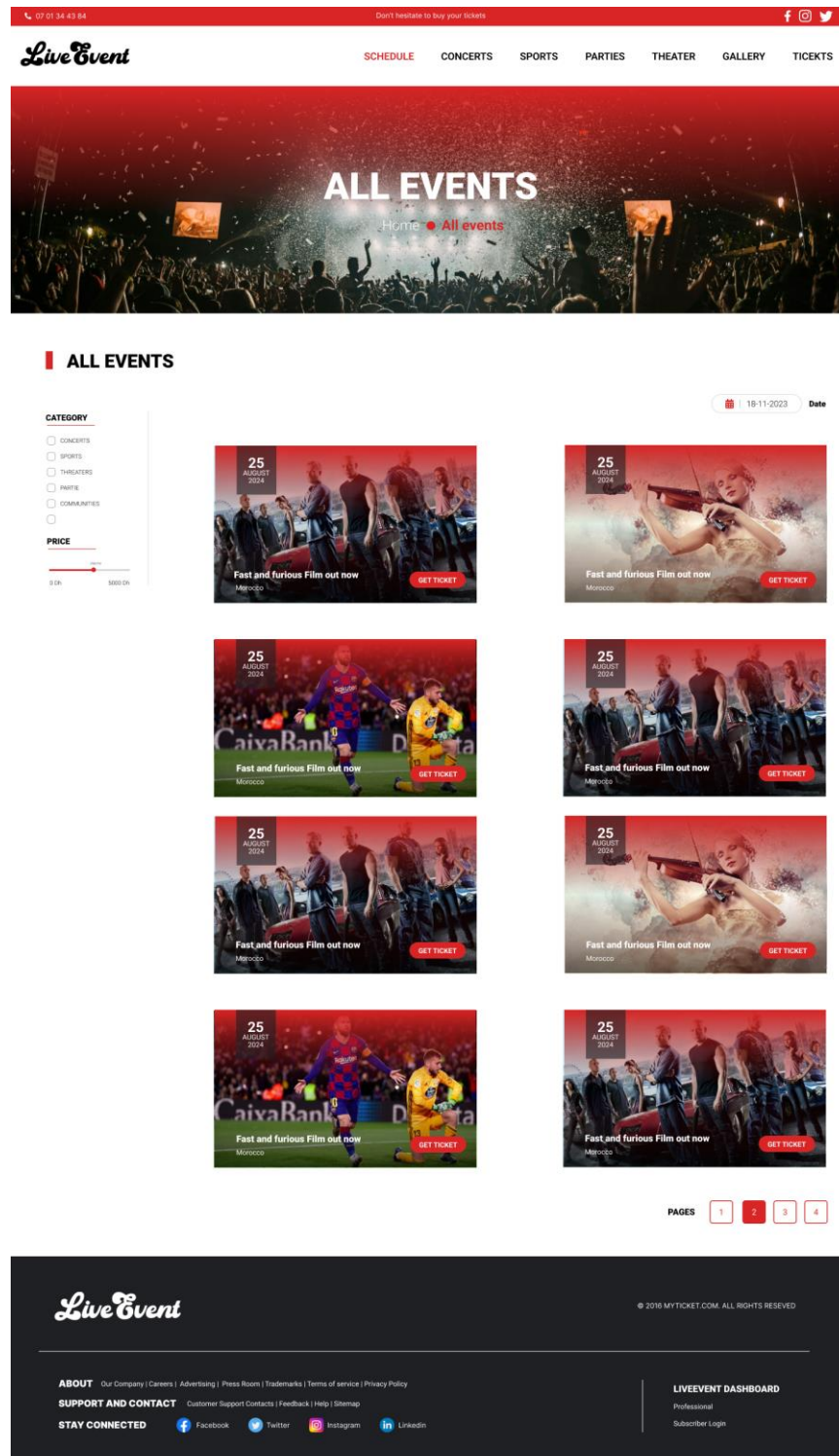


Figure 11 :Liste des Evènement

Cette interface permet aux utilisateurs de visualiser les événements qui existent et de les filtrer selon des critères tels que la date de l'événement, le prix du billet et la catégorie.

5.4 Interface Détail évènement :

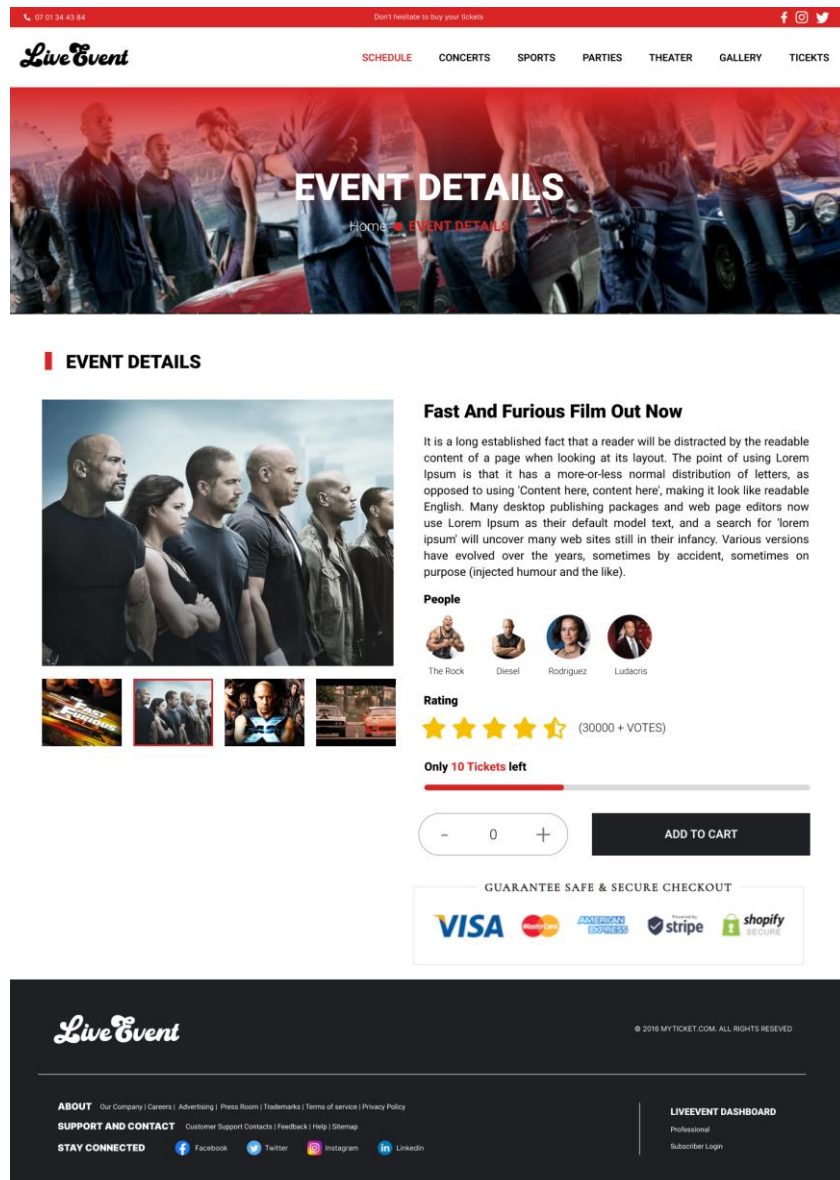


Figure 12 : Détail événement

Cette interface permet aux utilisateurs de consulter les détails d'un événement, le prix de l'événement, les personnes présentes dans l'événement. De plus, ils peuvent ajouter le billet à leur panier afin de réserver une place.

5.5 Interface Cart :

07 01 34 43 84



Don't hesitate to buy your tickets

f i t

LiveEvent

SCHEDULECONCERTSPORTSPARTIESTHEATERGALLERYTICKETS

Shopping Cart

Event	Quantity	Price	Total
 <div>Fast And Furious Film Out Now It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed</div>	- 1 +	250 Dh	250 Dh
 <div>FCB vs MAN CITY It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed</div>	- 3 +	1000 Dh	3000 Dh

You have a question ?

How can we help you ?

Coupon

* Discount will be calculated and applied at checkout

Coupon Code

Apply Coupon

3250 Dh

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed

I agree with the terms and conditions

Go back to events

CHECK OUT

LiveEvent

© 2016 MYTICKET.COM. ALL RIGHTS RESEVED

ABOUTOur Company | Careers | Advertising | Press Room | Trademarks | Terms of service | Privacy Policy

SUPPORT AND CONTACTCustomer Support Contacts | Feedback | Help | Sitemap

STAY CONNECTEDFacebookTwitterInstagramLinkedIn

LIVEEVENT DASHBOARD

Professional

Subscriber Login

Figure 13 : Interface Cart

Cette interface permet de consulter les événements ajoutés dans le panier, d'augmenter la quantité des billets voulus, et elle permet de calculer le total à payer pour les billets avant de passer au paiement.

5.6 Interface Paiement :

07 01 34 43 84

Don't hesitate to buy your tickets

f

@

t

LiveEvent

SCHEDULECONCERTSPORTSPARTIESTHEATERGALLERYTICKETS

Payment

Cart > Payment Method

Payment Information

Card Number

CVV

Expiration Date

VISA

MasterCard

AMERICAN EXPRESS


Powered by stripe

shopify SECURE

< Return to Cart

Vaild your payment


Your Events



Fast And Furious Film Out Now

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed

250 Dh



FCB vs MAN CITY

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed

3000 Dh

Total

3250 Dh

LiveEvent

© 2016 MYTICKET.COM. ALL RIGHTS RESEVED

ABOUT

Our Company | Careers | Advertising | Press Room | Trademarks | Terms of service | Privacy Policy

SUPPORT AND CONTACT

Customer Support Contacts | Feedback | Help | Sitemap

STAY CONNECTED

f

Facebook

t

Twitter

@

Instagram

in

LinkedIn

LIVEEVENT DASHBOARD

Professional

Subscriber Login

Figure 14 :Interface de paiement

Cette interface permet le paiement des billets à réserver pour les événements. Elle affiche le total à payer ainsi qu'un formulaire pour remplir les informations de la carte bancaire. Cette étape permet à l'utilisateur de réserver ses billets.

20

5.7 Interface liste des billets :

The screenshot displays the 'LiveEvent' website interface. At the top, there's a red header with the 'LiveEvent' logo, navigation links (SCHEDULE, CONCERTS, SPORTS, PARTIES, THEATER, GALLERY, TICKETS), and social media icons. Below the header is a large banner with the text 'YOUR TICKETS' and a background image of a crowd at a night event. A sub-header 'YOUR TICKETS' is followed by a list of three tickets. The first two tickets are for a football match 'FCBARCELONA - MANCHISTER CITY' on '12 NOV 2024'. The third ticket is for a film event 'FAST AND FURIOUS FILM' on the same date. Each ticket card includes event details, a price, a date, a time, and a barcode. The footer contains the 'LiveEvent' logo, copyright information, and links for 'ABOUT', 'SUPPORT AND CONTACT', 'STAY CONNECTED', and 'LIVEEVENT DASHBOARD'.

YOUR TICKETS

Event	Date	Time	Price	Gate	Seat
FCBARCELONA - MANCHISTER CITY	12 NOV 2024	09 PM	1000 Dh	99	0012
FCBARCELONA - MANCHISTER CITY	12 NOV 2024	09 PM	1000 Dh	99	0013
FAST AND FURIOUS FILM	12 NOV 2024	09 PM	250 Dh	01	0013

LiveEvent

© 2016 MYTICKET.COM. ALL RIGHTS RESEVED

ABOUT Our Company | Careers | Advertising | Press Room | Trademarks | Terms of service | Privacy Policy

SUPPORT AND CONTACT Customer Support | Contacts | Feedback | Help | Sitemap

STAY CONNECTED Facebook | Twitter | Instagram | LinkedIn

LIVEEVENT DASHBOARD Professional | Subscriber Login

Figure 15 : Interface liste des billets

Cette interface permet d'afficher tous les billets réservés par l'utilisateur pour accéder à l'événement.

Conclusion

En conclusion, revenons à la raison d'être de notre projet. Notre démarche visait à explorer, concevoir et mettre en œuvre une plateforme de réservation de billets pour divers événements, couvrant l'ensemble du cycle de développement, de la spécification des besoins à la mise en exploitation et au déploiement du système.

Ce projet symbolise non seulement la réalisation d'un processus de développement ambitieux, mais également une opportunité d'améliorer l'expérience des utilisateurs en matière de réservation d'événements. En mettant en lumière les réussites et les défis rencontrés tout au long du parcours, notre objectif est de proposer une plateforme qui transcende la simple fonctionnalité, incitant les utilisateurs à adopter une approche plus pratique et consciente lors de la participation à des événements divers.

Nous aspirons à ce que cette initiative ait un impact positif sur la facilité et la satisfaction des utilisateurs lors de la réservation d'événements, tout en démontrant la contribution bénéfique de la technologie moderne à l'amélioration de la qualité de vie et à la promotion d'une participation active à des expériences variées.