

Université de Sousse

Ecole Nationale d'Ingénieurs de Sousse



**Conception et mise en œuvre d'une plateforme automatisée
de scan et tests de sécurité basée sur des outils open source**

Réalisé par :

RAOUF BRAHEM, NAJD ELADEL & HOUSSEM EDDIN CHAOUACH

Encadré par :

Mme OLFA DALLEL

Département :

Informatique Industrielle

Groupe :

IA2.2

REMERCIEMENTS

À Nos Enseignants Exceptionnelles,

Mme. DALLEL OLFA & M. KHAYATI

NAOUFEL

C'est avec une immense gratitude que je prends la plume pour exprimer toute l'admiration que j'ai pour vous en tant qu'enseignants. Ce qui nous a le plus marqués, c'est votre bienveillance et votre soutien inconditionnel. Vous avez toujours été là pour répondre à nos questions, guider nos réflexions et encourager nos ambitions. Votre disponibilité constante et votre attention personnalisée ont fait de nous des étudiants confiants et motivés.

De la part de :

ELADEL Najd,

CHAOUACH Housseem Eddin

&

BRAHEM Raouf

Table Des Matières

REMERCIEMENTS	2
Table Des Matières.....	3
Liste Des Figures.....	5
Introduction Générale.....	6
Chapitre 1 : Environnement De Projet	7
Introduction	7
1.1. Présentation générale du projet	7
1.2. Problématique.....	7
1.3. Objectifs	7
1.4. Analyse de l'existant	8
1.4.1. Solutions existantes	8
1.4.2. Tableau comparatif.....	9
1.5. Prérequis et technologies utilisées.....	11
Conclusion.....	11
Chapitre 2 : Analyse et spécification des besoins	12
Introduction	12
2.1. Analyse des besoins	12
2.1.1. Principe.....	12
2.1.2. Identification des acteurs.....	12
2.2. Spécification des besoins	13
2.2.1. Principe.....	13
2.2.2. Les besoins fonctionnels par acteur	13
2.2.3. Les besoins non fonctionnels	15
2.2.4. Diagramme de cas d'utilisation général.....	15
Conclusion.....	16
Chapitre 3 : Conception.....	17
Introduction	17
3.1. Conception technique	17

3.1.1.	Diagramme de classes	17
3.1.2.	Diagrammes d'activité	18
Conclusion.....		20
Chapitre 4 : Réalisation & Simulation		21
Introduction		21
4.1.	Environnement de travail	21
4.1.1.	Environnement matériel	21
4.1.2.	Environnement logiciel	21
4.2.	Les interfaces.....	24
4.2.1.	Page principale	24
4.2.2.	Page connexion	24
4.2.3.	Simulation de l'outil « SUBFINDER ».....	25
Conclusion.....		25
Conclusion Générale		26
Bibliographie & Néographie		27

Liste Des Figures

Figure 1 : Pesntest-tools.com platform.....	8
Figure 2 : Imtruder platform	9
Figure 3 : Les besoins non fonctionnels.....	15
Figure 4 : Diagramme de cas d'utilisation générale.....	16
Figure 5 : Logo StarUML	17
Figure 6 : Diagramme de classes.....	18
Figure 7 : Diagramme d'activité "Authentication"	18
Figure 8 : Diagramme d'activité "Inscription"	19
Figure 9 : Logo VS Code.....	21
Figure 10 : Logo XAMPP	22
Figure 11 : Logo MySQL.....	22
Figure 12 : Logo HTML5	22
Figure 13 : Logo CSS3	23
Figure 14 : Logo JavaScript	23
Figure 15 : Logo Flask	23
Figure 16 : Page principale.....	24
Figure 17 : Page connexion.....	24
Figure 18 : Simulation de l'outil "SUBFINDER"	25

Introduction Générale

Le commerce électronique, ou e-commerce, est un secteur dynamique en pleine croissance, permettant l'achat et la vente de biens et services en ligne. Il implique la commercialisation de produits via des sites web ou des applications mobiles, avec des solutions de paiement en ligne sécurisées. L'e-commerce a connu une expansion rapide ces dernières années, principalement en raison de l'augmentation du nombre de consommateurs effectuant des achats en ligne.

Les avantages pour les consommateurs sont multiples, offrant la possibilité de faire des achats à tout moment et depuis n'importe où, sans avoir à se déplacer physiquement en magasin. De plus, il facilite la comparaison des prix et des caractéristiques des produits, tout en offrant une plus grande variété d'articles.

Notre initiative vise à simplifier l'expérience d'achat de vêtements en rassemblant des marques renommées telles que ZARA, H&M, SHEIN, NIKE, etc...., offrant une gamme complète d'articles à des prix raisonnables. Nous nous engageons à garantir une qualité exceptionnelle, en excluant toute marque non authentique, pour assurer aux consommateurs l'accès à une diversité de produits de qualité tout en préservant l'authenticité des marques proposées.

Le premier chapitre de ce rapport initie une présentation globale du projet, exposant l'idée principale. Enfin, des solutions sont formulées pour anticiper les problèmes identifiés.

Le deuxième chapitre, intitulé "Analyse et Spécification des besoins", se concentre sur les exigences fondamentales du site. Une analyse approfondie, couvrant à la fois les besoins fonctionnels et non fonctionnels, est réalisée. Cela positionne notre site de manière adéquate dans son contexte.

Dans le troisième chapitre, "Conception", nous détaillons la méthodologie spécifique choisie pour la conception. Nous mettons en lumière les acteurs impliqués et présentons divers diagrammes, dont ceux de classes, d'activité, et de la base de données du site. Des explications détaillées sont fournies pour chacun de ces éléments.

Le dernier chapitre, "Réalisation & Simulation", expose les outils de développement utilisés, les principales interfaces graphiques créées, ainsi que quelques scénarios applicatifs.

Chapitre 1 : Environnement De Projet

Introduction

Ce chapitre aura pour objectif de dresser la présentation générale de notre projet, en abordant la problématique, les objectifs visés, ainsi que les outils qui seront utilisés pour sa réalisation et son implémentation.

1.1. Présentation générale du projet

Ce projet vise à concevoir une plateforme automatisée de scan et de tests de sécurité utilisant des outils open source pour **détecter efficacement les vulnérabilités**. Il répond au besoin croissant d'accessibilité et d'efficacité dans la cybersécurité, notamment pour les PME. En centralisant et standardisant les processus, il réduit les coûts, améliore la rapidité d'analyse et renforce la sécurité des systèmes.

1.2. Problématique

Dans un contexte où les attaques informatiques deviennent de plus en plus sophistiquées et fréquentes, les entreprises ont besoin de solutions rapides, précises et adaptées pour protéger leurs systèmes. Les solutions de sécurité commerciales, bien que puissantes, sont souvent coûteuses et limitées en personnalisation. De plus, les tests manuels demandent une expertise avancée et sont difficilement reproductibles à grande échelle.

1.3. Objectifs

Ce projet a pour objectif de concevoir et de déployer une plateforme intégrée pour automatiser les processus de reconnaissance, de scan de vulnérabilités et de tests de sécurité. En tant qu'élèves ingénieurs, nous explorerons des outils open source et documenterons les workflows pour en assurer le maintien. Ce projet répond à ces défis en proposant une plateforme automatisée basée sur des outils open source, combinant des technologies modernes pour :

- **Automatiser la détection des vulnérabilités** grâce à des scans réguliers et ciblés, réduisant le temps d'analyse par rapport aux méthodes traditionnelles ;
- **Intégrer des outils éprouvés et modulables** pour un coût réduit, tout en permettant une adaptation rapide aux besoins spécifiques ;

- **Standardiser et centraliser les processus de test**, garantissant une méthodologie reproductible et une couverture exhaustive des menaces potentielles.

1.4. Analyse de l'existant

1.4.1. Solutions existantes

1.4.1.1. Pentest-tools.com

Obtenez la perspective d'un hacker sur vos applications web, votre réseau et votre cloud. Pentest-Tools.com aide les équipes de sécurité à réaliser les étapes clés d'un test de pénétration, facilement et sans compétences spécialisées en hacking. Basé en Europe (à Bucarest, Roumanie), **Pentest-Tools.com** propose des outils de cybersécurité offensifs et des logiciels de scanner de vulnérabilités propriétaires pour les testeurs de pénétration et autres professionnels de la sécurité informatique. Les équipes de sécurité utilisent notre boîte à outils pour identifier les chemins que les attaquants peuvent emprunter pour compromettre votre organisation, afin que vous puissiez réduire efficacement votre exposition aux cyberattaques.

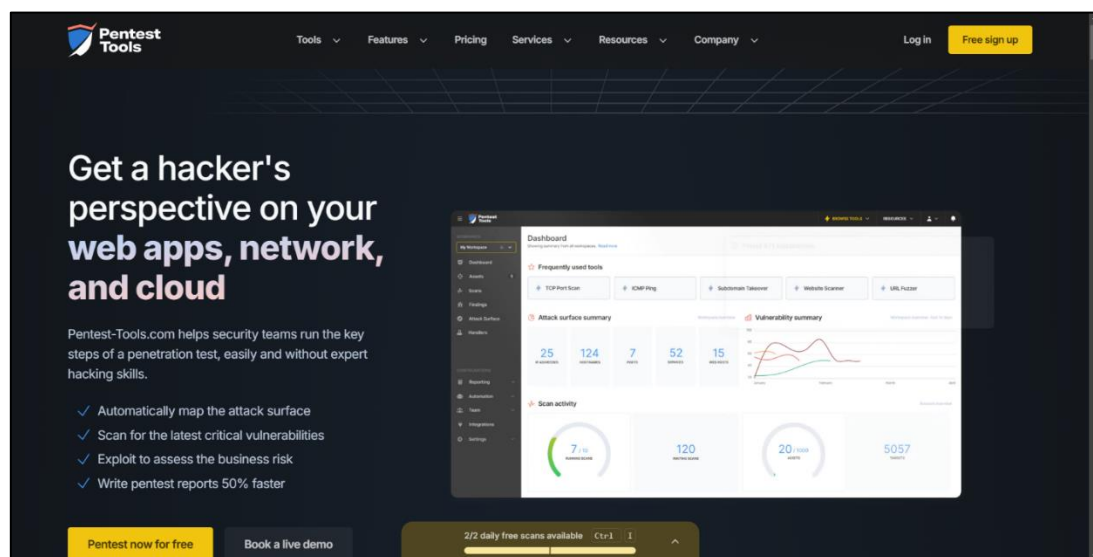


Figure 1 : Pentest-tools.com platform

1.4.1.2. Intruder

Intruder est une plateforme de gestion des vulnérabilités qui offre une vue en temps réel des surfaces d'attaque grâce à une surveillance continue du réseau, un scan automatisé des vulnérabilités et une réponse proactive aux menaces. Avec des rapports prêts pour les audits et des résultats exploitables priorisés par contexte, **Intruder** aide plus de 2 500 clients à travers le monde à se concentrer sur la correction des problèmes essentiels, rendant la gestion des

vulnérabilités facile et efficace. **Intruder** s'intègre à AWS, Azure, Google Cloud, Slack, Jira, et bien plus encore.

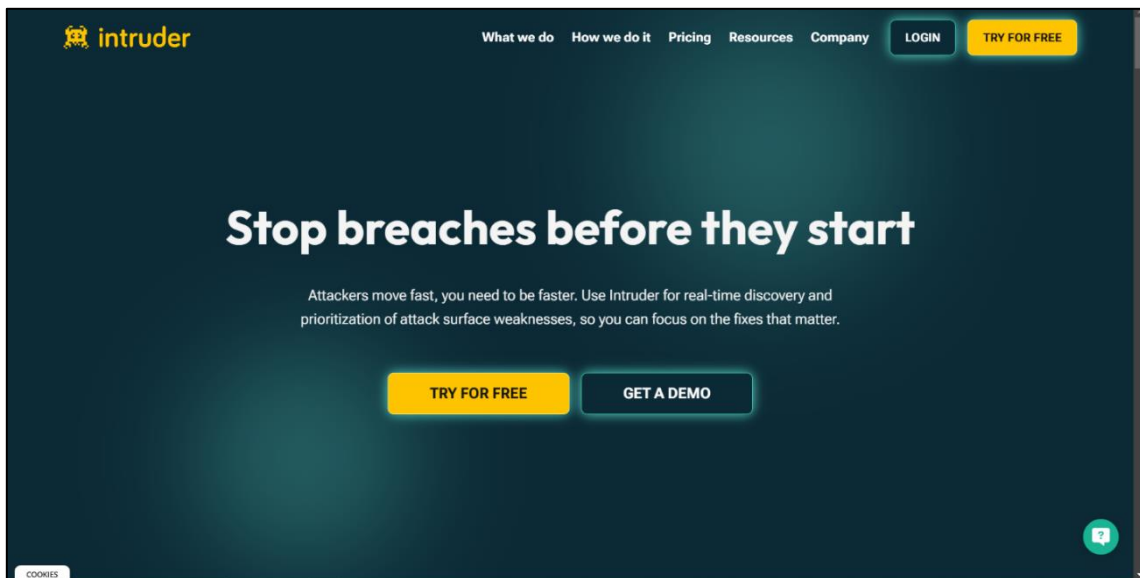


Figure 2 : Intruder platform

1.4.2. Tableau comparatif

Ce tableau compare les deux plateformes, **Intruder** et **Pentest-Tools.com**, en termes de fonctionnalités, avantages et désavantages. Il met en évidence leurs principales différences, telles que la gestion continue des vulnérabilités et la flexibilité des tests de pénétration. Ce comparatif permet de mieux comprendre quel outil convient selon les besoins spécifiques en cybersécurité.

Critère	Intruder	Pentest-tools
Type de Plateforme	Gestion des vulnérabilités avec tests automatisés	Tests de pénétration automatisés
Fonctionnalités	Surveillance continue du réseau, scan de vulnérabilités automatisé, réponse proactive aux menaces	Outils pour la reconnaissance, scan de vulnérabilités, tests de pénétration
Cible	Entreprises cherchant une gestion continue des vulnérabilités	Utilisateurs cherchant à effectuer des tests de pénétration ponctuels
Rapports	Rapports prêts pour les audits, résultats exploitables priorisés par contexte	Rapports détaillés avec des suggestions de correction
Intégrations	AWS, Azure, Google Cloud, Slack, Jira, etc.	Intégrations limitées, mais outils compatibles avec diverses plateformes
Facilité d'utilisation	Interface intuitive, principalement automatisée	Interface plus technique, adaptée aux utilisateurs expérimentés
Public Cible	PME et grandes entreprises cherchant une gestion proactive de la sécurité	Pentesters et professionnels de la cybersécurité
Méthodologie de Test	Scans réguliers et tests automatisés	Tests ponctuels via divers outils d'attaque
Accès aux Outils	Accès complet via abonnement, plan basé sur les besoins	Abonnement basé sur l'accès aux outils spécifiques
Rapidité d'Analyse	Temps d'analyse en temps réel, continuellement mis à jour	Temps d'analyse plus long, dépend des tests manuels
Tarification	Abonnement mensuel basé sur les services choisis	Abonnement à la carte en fonction des outils utilisés
Avantages	<ul style="list-style-type: none"> - Surveillance continue et proactive des vulnérabilités - Rapports prêts pour les audits - Intégration avec des outils populaires (AWS, Azure, Slack, Jira) 	<ul style="list-style-type: none"> - Outils spécifiques pour divers tests de pénétration - Grande flexibilité dans le choix des outils - Convient aux pentesters expérimentés
Désavantage	<ul style="list-style-type: none"> - Moins adapté pour les tests ponctuels - Moins de contrôle sur les tests manuels - Coût récurrent basé sur l'abonnement mensuel 	<ul style="list-style-type: none"> - Moins de gestion continue des vulnérabilités - Interface plus complexe et moins automatisée - Nécessite une expertise technique plus élevée

Table 1 : Comparaison entre Intruder et Pentest-Tools.com

1.5. Prérequis et technologies utilisées

Développement **FRONTEND** : HTML5, CSS3, JAVASCRIPT.

Développement **BACKEND** : Flask, MySQL.

Conclusion

En conclusion, ce chapitre a permis de poser les bases fondamentales de notre projet en présentant de manière claire la problématique à laquelle nous répondons, les objectifs que nous visons ainsi que les outils qui seront mobilisés pour mener à bien sa réalisation et son implémentation. Cette introduction prépare le terrain pour les développements détaillés qui suivront, en mettant en lumière les enjeux et les choix technologiques qui orienteront le projet dans les étapes suivantes.

Chapitre 2 : Analyse et spécification des besoins

Introduction

La phase de spécification est cruciale pour l'organisation efficace de notre projet avant le développement du site web. Ce chapitre, faisant suite à la présentation initiale, se concentre sur les besoins fonctionnels et non fonctionnels ainsi que sur les acteurs du site web. Pour plus de clarté, un diagramme de cas d'utilisation sera également présenté.

2.1. Analyse des besoins

2.1.1. Principe

L'analyse des besoins pour un site web implique une compréhension approfondie des objectifs, du public cible, des fonctionnalités essentielles, et de l'expérience utilisateur recherchée. En déterminant ces éléments clés, on établit les bases pour le choix des technologies, la conception du contenu, et l'optimisation tant du point de vue de la performance que de la sécurité. Cette démarche garantit la création d'un site web qui répond pleinement aux attentes des utilisateurs tout en soutenant efficacement les objectifs de l'entreprise.

2.1.2. Identification des acteurs

Dans cette partie, nous procédons à l'identification des intervenants qui interagiront avec le système. Un intervenant peut être une personne ou un logiciel représentant le rôle assumé par des entités externes telles que des utilisateurs, des dispositifs matériels, ou d'autres systèmes, interagissant directement avec le système en question. L'intervenant est une abstraction de ces entités et revêt une importance capitale dans les interactions avec le système.

2.1.2.1. Administrateur (Admin)

Dans le contexte des sites web, le terme fréquemment utilisé pour décrire cette fonction est celui de « **webmaster** ». Le webmaster assume la responsabilité de maintenir la dynamique du site et de garantir les mises à jour des outils, ainsi que de superviser la gestion des utilisateurs.

En tant qu'acteur central, le webmaster possède la capacité d'accomplir toutes les tâches associées au site, y compris l'accès aux statistiques de toutes les pages et la consultation des détails des pages de chaque client.

2.1.2.2. Client

Le client utilise la plateforme pour initier des scans de sécurité, configurer des tests automatisés et analyser les rapports générés. Il exploite ces résultats pour corriger les vulnérabilités et améliorer la sécurité de son infrastructure. Il est important de souligner que notre système assume entièrement la responsabilité de la confidentialité des données personnelles de nos clients.

2.2. Spécification des besoins

2.2.1. Principe

Le document de spécification des besoins constitue un recueil complet regroupant les besoins fonctionnels, non fonctionnels, et la planification prévue. En tant qu'accord contractuel entre le client et l'équipe de développement, il doit être détaillé pour éviter tout litige ou retard. Les informations qu'il contient sont suffisamment précises pour permettre à une nouvelle équipe de commencer le codage directement après sa consultation, décrivant en détail le développement prévu. Il revêt une importance cruciale, assurant une vision détaillée du projet, avec des phases préliminaires déjà complétées pour garantir la faisabilité et les choix technologiques. En intégrant une étude de l'existant, il évite la redondance et offre une conception générale du projet en cas de reprise.

2.2.2. Les besoins fonctionnels par acteur

Les besoins fonctionnels définissent les actions que le système doit entreprendre en réponse à une sollicitation. Notre site web doit répondre à ces besoins en assurant une expérience utilisateur optimale. Ces exigences incluent la navigation fluide, la gestion de contenu efficace, des fonctionnalités de recherche performantes, et une interface intuitive pour les utilisateurs.

Notre site web doit satisfaire les exigences fonctionnelles énumérées dans la partie suivante.

2.2.2.1. Administrateur

➤ Gérer les clients :

- **Enregistrement des clients** : Procédure d'inscription et de création de compte pour les utilisateurs du site.
- **Administration des profils clients** : Offre aux clients la possibilité de gérer leurs informations personnelles, telles que les coordonnées et les préférences.
- **Authentification et sécurité** : Utilisation de mécanismes d'authentification pour garantir la sécurité de l'accès au compte des clients et la protection de leurs données.
- **Suivi des activités des clients** : Gérer les logs et assurer la traçabilité des actions effectuées.

➤ Gestion des outils : Configurer et mettre à jour les outils de scan (Subfinder, Nmap, OWASP ZAP, Gophish, etc.).

➤ Service client : Établissement d'un système de support permettant aux clients de poser des questions, de soumettre des réclamations ou d'obtenir de l'aide en cas de problème.

➤ Surveiller l'état des serveurs et des scans en cours.

➤ Modifier les paramètres de sécurité de la plateforme.

➤ Superviser l'envoi des notifications et alertes aux clients.

➤ Effectuer des mises à jour et maintenances techniques.

➤ Détecter et gérer les abus ou utilisations malveillantes du système.

2.2.2.2. Client

➤ Gérer le compte :

- **Créer un compte** : S'inscrire sur le site pour accéder à ses fonctionnalités.
- **Se connecter** : Accéder à un compte existant en fournissant les informations d'identification.
- **Gérer les informations personnelles** : Mettre à jour les données personnelles liées au compte.
- **Préférences et notifications** : Personnaliser les préférences de communication et les notifications.
- **Réinitialiser le mot de passe** : Réinitialiser le mot de passe en cas d'oubli ou de nécessité.

- **Historique des tests** : Consulter l'historique détaillé des tests réalisées.
- **Supprimer le compte** : Offrir la possibilité de supprimer définitivement le compte utilisateur si nécessaire.

2.2.3. Les besoins non fonctionnels



Figure 3 : Les besoins non fonctionnels

2.2.4. Diagramme de cas d'utilisation général

Pour synthétiser l'ensemble des fonctionnalités discutées, nous pouvons maintenant passer à la création du diagramme de cas d'utilisation générale illustré ci-dessous.

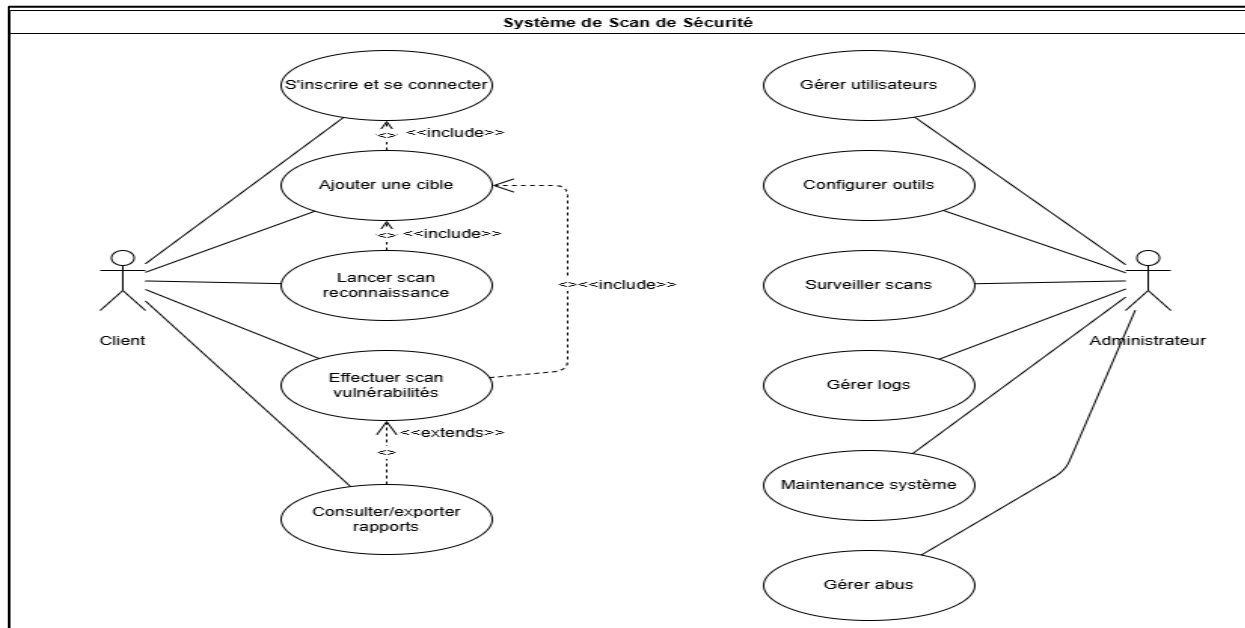


Figure 4 : Diagramme de cas d'utilisation générale

Conclusion

Dans ce chapitre, une analyse approfondie des besoins du projet a été menée, mettant en lumière l'état actuel. Dans le prochain chapitre, nous explorerons la phase conceptuelle, fondement de la future conception du site.

Chapitre 3 : Conception

Introduction

La phase de conception revêt une importance cruciale dans le processus de développement du site web. Nous mettrons un accent particulier sur l'aspect visuel, tout en approfondissant la composante technique à travers l'élaboration des autres diagrammes essentiels.

3.1. Conception technique

UML propose une approche astucieuse pour illustrer diverses perspectives d'un système à travers des vues. Une vue se compose d'un ou plusieurs diagrammes, avec deux types principaux : la vue statique, dépeignant le système physiquement, et la vue dynamique, révélant son fonctionnement. Cette section introduira le diagramme de classes, une vue statique de notre application, Créés à l'aide du logiciel StarUML.



Figure 5 : Logo StarUML

3.1.1. Diagramme de classes

Les diagrammes de classes sont employés pour représenter les composants statiques d'un modèle, incluant les classes et les paquets, afin de visualiser leur structure. Dans notre contexte, la figure ci-dessous présente de manière globale le diagramme de classes de notre site.

Exemples :

- L'administrateur peut **gérer** 1 ou plusieurs clients.
- Un client peut **effectuer** 1 ou plusieurs tests.
- Un rapport peut contenir 1 ou plusieurs tests.

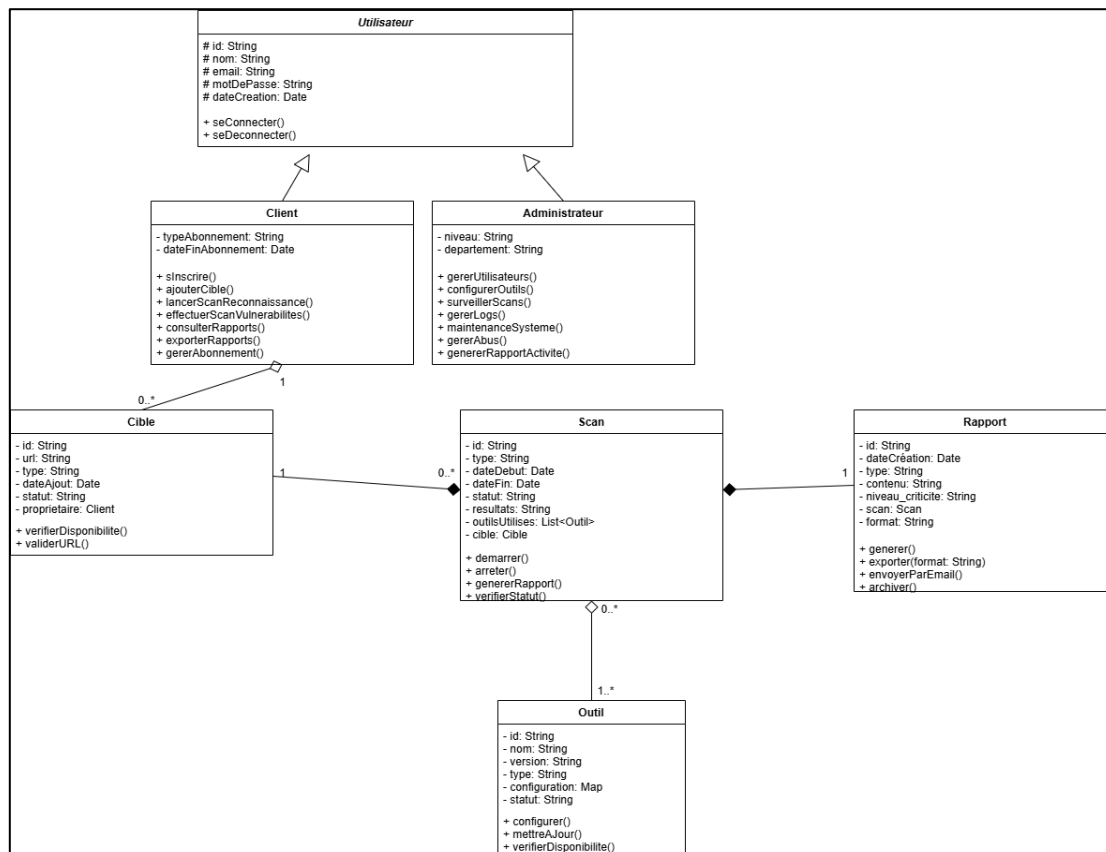


Figure 6 : Diagramme de classes

3.1.2. Diagrammes d'activité

Les diagrammes d'activité, dans notre contexte, servent à illustrer de manière visuelle les processus et les interactions associés aux fonctionnalités du site. Ils offrent une représentation claire des différentes étapes et actions entreprises par les utilisateurs, facilitant ainsi la compréhension des opérations liées au commerce électronique.

3.1.2.1. Diagramme d'activité « Authentification »

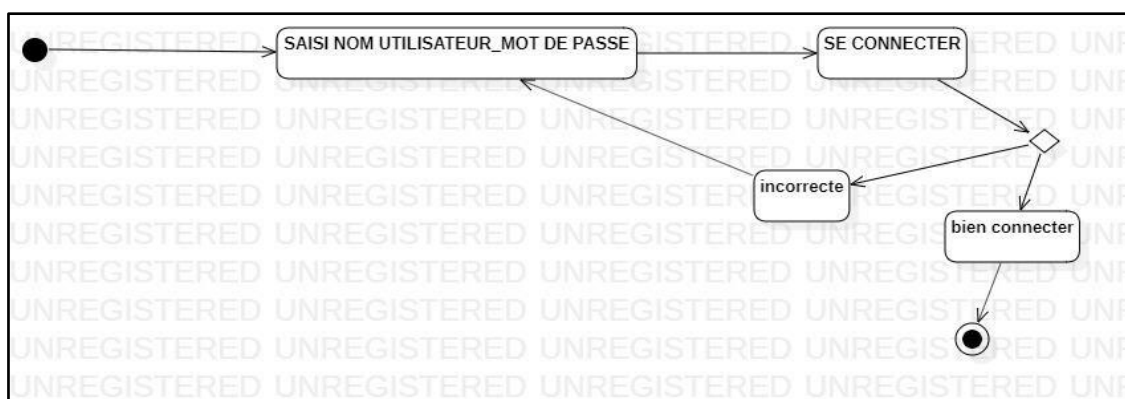


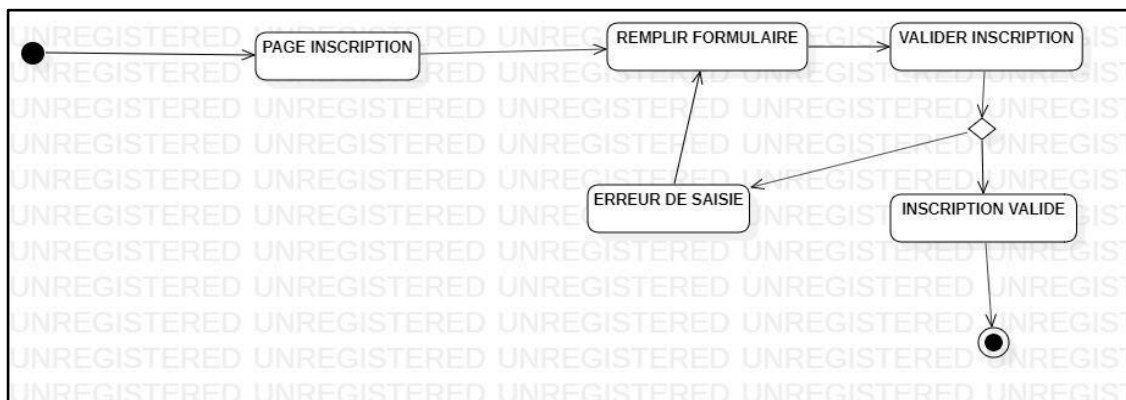
Figure 7 : Diagramme d'activité « Authentification »

➤ **Scénarios :**

- **Titre :** Authentification
- **Objectif :** Il s'agit de l'activité principale où l'utilisateur doit s'authentifier.
- **Acteurs :** Administrateur et le client.
- **Préconditions :** L'utilisateur doit procéder à l'authentification.

➤ **Enchaînements :**

- Le système demande à l'utilisateur de s'authentifier.
- L'utilisateur saisit son identifiant et son mot de passe.
- Si la saisie est correcte, le système affiche la page suivante.
- En cas d'erreur, le système affiche un message indiquant "Identifiant ou mot de passe incorrect".

3.1.2.2. Diagramme d'activité « Inscription »*Figure 8 : Diagramme d'activité "Inscription"*➤ **Scénarios :**

- **Titre :** Inscription
- **Objectif :** Le visiteur doit s'inscrire.
- **Acteur :** Visiteur
- **Préconditions :** Le visiteur souhaite s'inscrire.

➤ **Enchaînements :**

- Le visiteur remplit le formulaire d'inscription.
- Le système valide l'inscription.
- En cas de saisie correcte, le système affiche la page suivante.
- En cas d'erreur, le système affiche un message demandant de remplir tous les champs.

Conclusion

Au cours de cette section, nous avons élaboré notre système à partir de diagrammes de classes et d'activités. La prochaine étape consistera à explorer la mise en œuvre et la simulation de notre site web. Ce passage marque la transition vers la mise en œuvre pratique des concepts élaborés dans la phase précédente.

Chapitre 4 : Réalisation & Simulation

Introduction

Après avoir finalisé l'étude et la conception de notre site, nous passons à la construction et à l'implémentation. Ce chapitre résume les résultats du projet, présentant brièvement l'environnement matériel et logiciel, avec quelques captures d'écran illustrant les principales fonctionnalités.

4.1. Environnement de travail

4.1.1. Environnement matériel

Pour la mise en œuvre de notre projet, nous utilisons un ordinateur portable doté des spécifications suivantes :

- Modèle : ACER
- Système d'exploitation : Windows 10 Professionnel 64 bits
- Processeur : Intel Core i5
- Mémoire : 8 Go de RAM

4.1.2. Environnement logiciel

Pour le développement de notre site, nous avons fait usage des logiciels suivants :

4.1.2.1. VS Code (version 1.85)

Visual Studio Code, éditeur de code extensible de Microsoft, est populaire pour ses fonctionnalités de débogage, IntelliSense et intégration Git. Bien que son code source soit open source, les binaires compilés sont en freeware, et il a été classé comme l'IDE le plus utilisé en 2023 selon un sondage Stack Overflow.



Figure 9 : Logo VS Code

4.1.2.2. XAMPP

XAMPP est une solution de pile logicielle serveur web gratuite et open source, développée par Apache Friends. Elle se compose principalement du serveur HTTP Apache, de la base de données MariaDB, et des interprètes pour les scripts écrits dans les langages de programmation PHP et Perl.



Figure 10 : Logo XAMPP

4.1.2.3. MySQL

MySQL est un système de gestion de base de données relationnelles en open source. Son nom est une combinaison de "My", le prénom de la fille du co-fondateur Michael Widenius, My, et "SQL", l'acronyme de Structured Query Language.



Figure 11 : Logo MySQL

4.1.2.4. HTML5

HTML5 est un langage de balisage utilisé pour structurer et présenter du contenu sur le World Wide Web. C'est la cinquième et dernière version majeure d'HTML recommandée par le World Wide Web Consortium. La spécification actuelle est connue sous le nom de HTML Living Standard.



Figure 12 : Logo HTML5

4.1.2.5. CSS3

Les feuilles de style en cascade (CSS) sont un langage de feuille de style utilisé pour spécifier la présentation et le style d'un document rédigé dans un langage de balisage tel que HTML ou XML. Le CSS est une technologie fondamentale du World Wide Web, aux côtés de HTML et JavaScript.



Figure 13 : Logo CSS3

4.1.2.6. JavaScript

JavaScript, souvent abrégé en JS, est un langage de programmation et une technologie fondamentale du World Wide Web, aux côtés de HTML et CSS. En 2023, 98,7 % des sites web utilisent JavaScript côté client pour le comportement des pages web, intégrant fréquemment des bibliothèques tierces.



Figure 14 : Logo JavaScript

4.1.2.7. Flask

Flask est un micro framework open-source de développement web en Python. Il est classé comme microframework car il est très léger. Flask a pour objectif de garder un noyau simple mais extensible. Il n'intègre pas de système d'authentification, pas de couche d'abstraction de base de données, ni d'outil de validation de formulaires. Cependant, de nombreuses extensions permettent d'ajouter facilement des fonctionnalités. Il est distribué sous licence BSD.



Figure 15 : Logo Flask

4.2. Les interfaces

4.2.1. Page principale

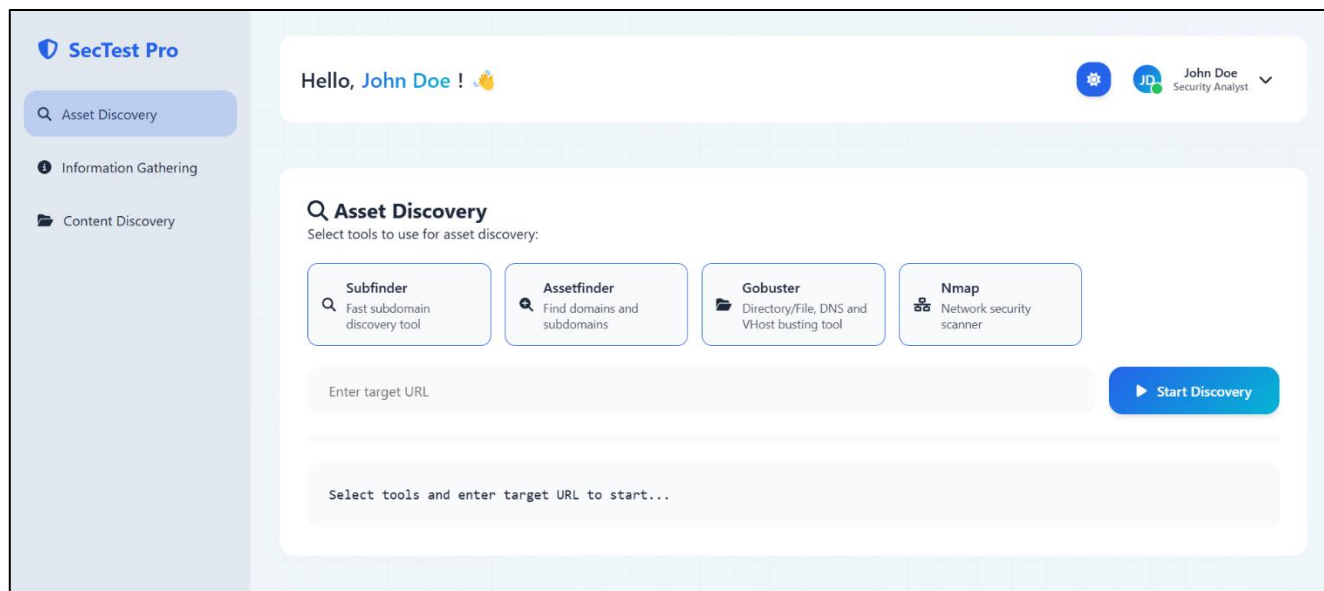


Figure 16 : Page principale

4.2.2. Page connexion

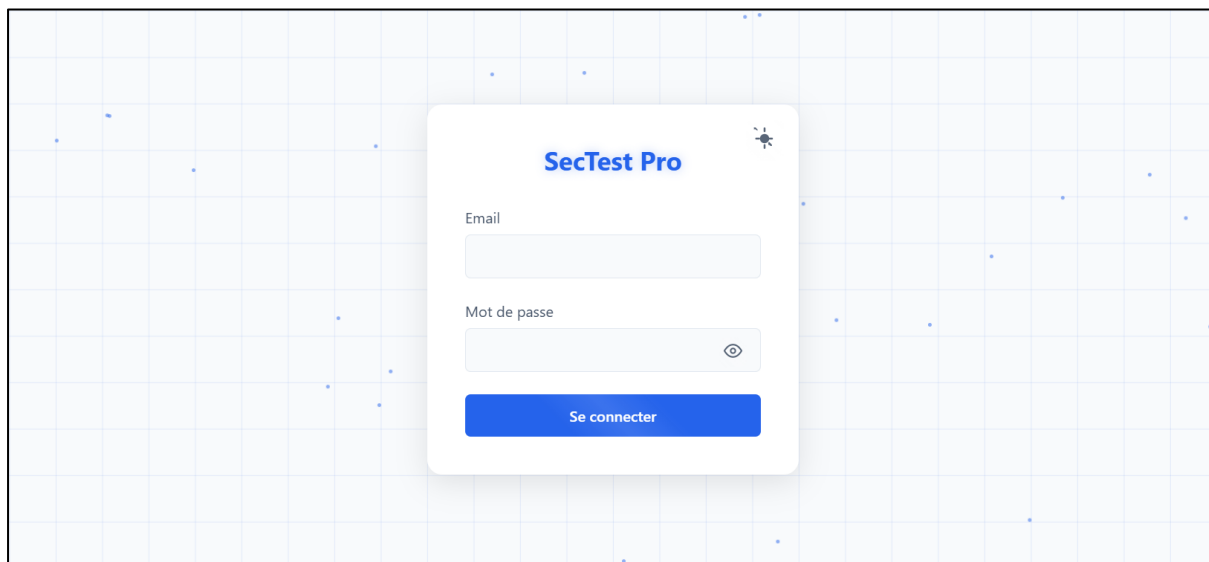


Figure 17 : Page connexion

4.2.3. Simulation de l'outil « SUBFINDER »

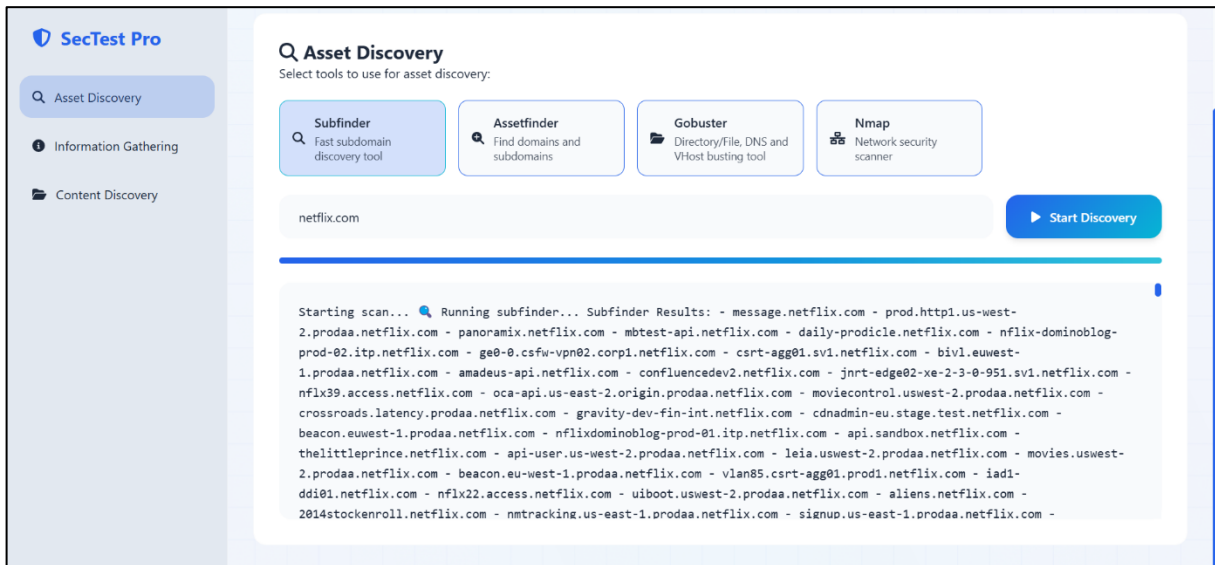


Figure 18 : Simulation de l'outil "SUBFINDER"

Conclusion

Ce chapitre détaille la phase de réalisation de notre site, la plus étendue du développement, s'étendant sur plus de trois mois pour atteindre nos objectifs fixés. Le résultat final est un produit opérationnel, conforme aux fonctionnalités des cas d'utilisation définis précédemment, prêt à être déployé pour les utilisateurs.

Conclusion Générale

L'objectif de ce projet innovant est de développer une plateforme web dédiée aux tests de pénétration automatisés, offrant une solution complète et accessible pour évaluer la sécurité des systèmes informatiques. Ce rapport présente un aperçu détaillé de notre démarche, de la conception initiale jusqu'à la réalisation finale de notre solution.

Dans la création de la plateforme "**SecTest Pro**", notre approche a débuté par une analyse approfondie des besoins en matière de cybersécurité et des solutions existantes sur le marché. Cette phase préliminaire nous a permis d'identifier les fonctionnalités essentielles et les points différenciants de notre solution. Tout au long du développement, nous avons mis en œuvre nos compétences en sécurité informatique et en développement web, renforçant ainsi notre expertise dans ces domaines cruciaux.

Notre démarche s'est focalisée sur la compréhension des besoins spécifiques des professionnels de la sécurité et des entreprises souhaitant évaluer leurs systèmes. Cette expérience nous a permis de développer une solution robuste et intuitive, tout en relevant les défis techniques complexes inhérents à ce type de projet.

La réalisation de la plateforme a nécessité l'intégration de multiples outils de sécurité reconnus (Subfinder, Nmap, OWASP ZAP, Gophish) et l'utilisation de technologies modernes pour le développement web, démontrant notre capacité à orchestrer des solutions techniques sophistiquées tout en maintenant une interface utilisateur accessible.

Les perspectives d'évolution de notre plateforme sont nombreuses. Nous envisageons d'intégrer des fonctionnalités d'intelligence artificielle pour l'analyse automatisée des vulnérabilités, d'étendre la gamme d'outils de sécurité disponibles, et d'implémenter un système de reporting plus avancé avec des recommandations personnalisées.

En conclusion, ce projet a considérablement enrichi notre expertise dans le domaine de la cybersécurité et du développement de solutions web complexes. Il représente la convergence de nos connaissances techniques et de notre vision d'une sécurité informatique accessible et automatisée.

Bibliographie & Néographie

Lien 1 : <https://www.wikipedia.org>

Lien 2 : <https://flask.palletsprojects.com>

Lien 3 : <https://www.w3schools.com>