

Problème : surcharge informatique

EQUIPE DASH-N

Membres de l'équipe:

- Amedjkouh Abdelghani
- Henry Crann
- Sophie Mason
- Dagobert Fotso
- Nice Umba



Agenda

1. Résumer
2. Introduction
3. Diagramme de gant
4. Processus de conception et informations supplémentaires
5. Prototype I
 - Résultat
 - test
 - amélioration
7. Prototype II
8. Prototype III complet
9. Conclusion (conclusion)
10. Questions



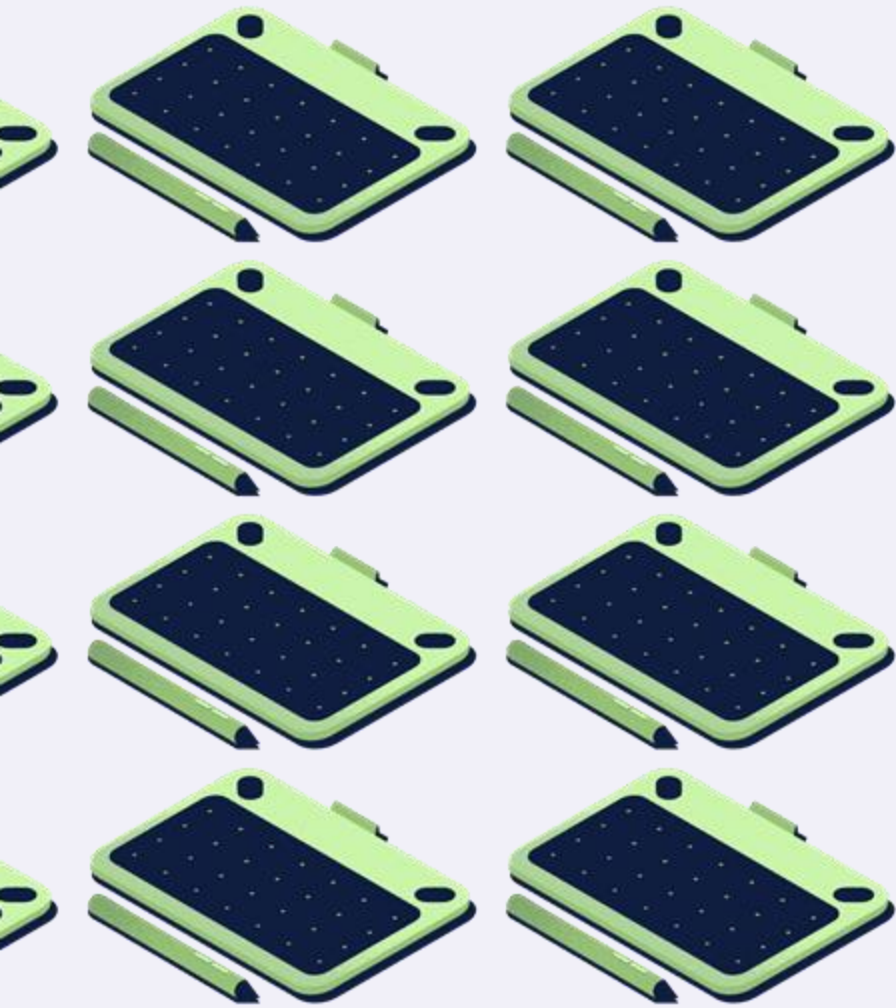


1. RESUMER

Cette présentation va résumer et expliquer notre processus de conception depuis le début et les étapes que nous avons suivies afin de réaliser notre produit final. Nous allons couvrir les étapes générales que nous avons suivies ainsi, et comment elles sont liées au processus de conception. Nous allons aussi donner une explication de chaque prototype et des tests et recherches et recherche qui nous ont aidé avec.


2.Introduction

Au sein d'un contexte professionnel en mouvement, il est crucial d'avoir une communication efficace afin de garantir le bon déroulement des activités et la productivité des équipes. Néanmoins, le trop plein d'informations peut affecter la productivité, provoquer des distractions et compliquer la gestion de la communication interne. Pour répondre à cette problématique, nous avons créé une solution novatrice : DASH-N, une application qui facilite la diffusion d'informations essentielles sans interrompre les activités des travailleurs. Ce programme utilise des notifications intelligentes pour trier les messages par importance, garantissant que tous les employés reçoivent les informations essentielles au moment opportun sans perturber leur travail. DASH-N assure une communication efficace en milieu professionnel, tout en renforçant la réactivité et la coopération inter-équipes. Au cours de cette présentation, nous allons vous montrer la façon dont DASH-N modifie la circulation des informations en entreprise et pourquoi il est crucial pour surmonter les obstacles de communication interne.

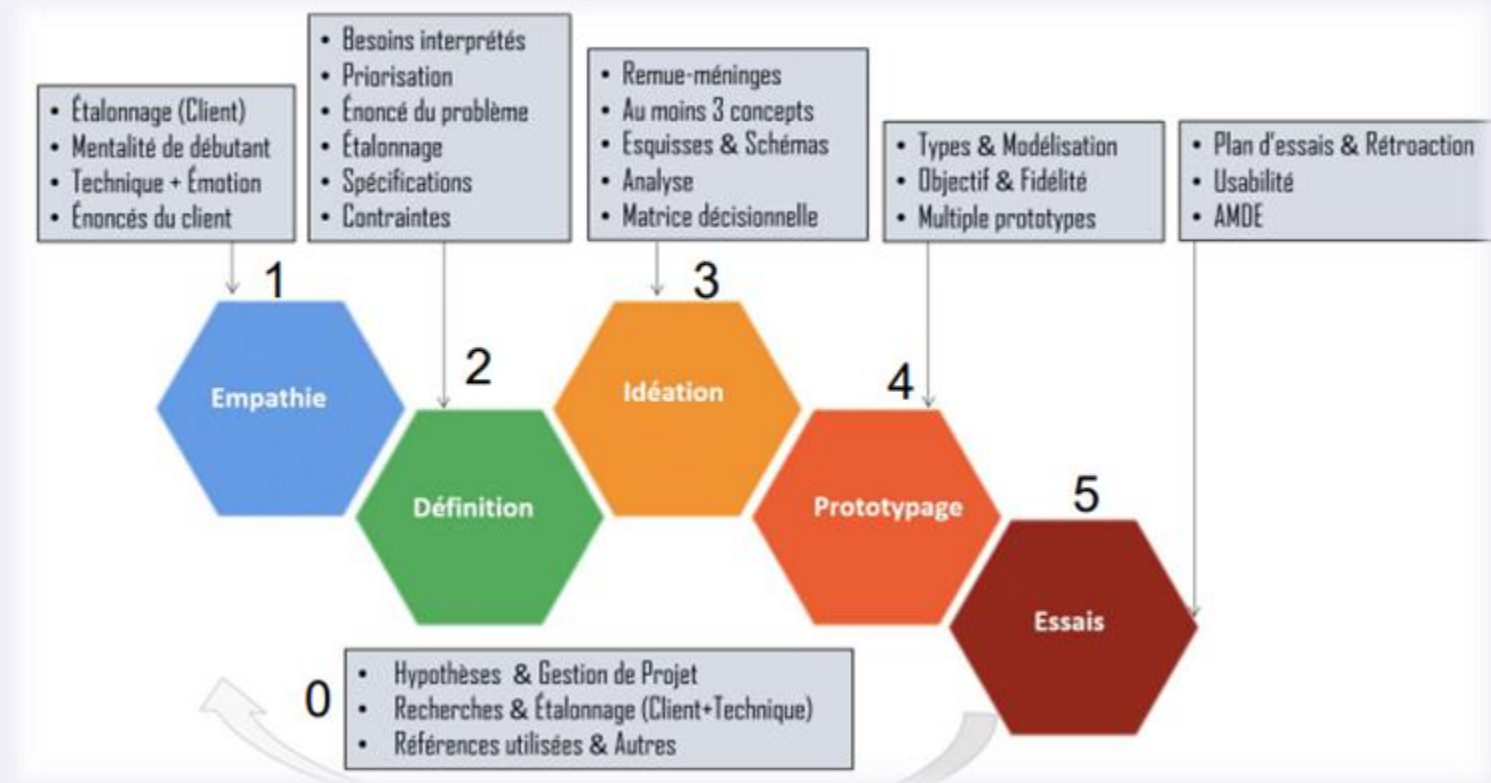


3. Diagram de Gantt:

	Septembre			Octobre				Novembre			
Tâches	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11
Formation d'équipe et contrat de projet											
Besoins du client et énoncé du problème											
Critères de conception											
Conceptualisation											
Plan et coût du projet											
Prototype I											
Prototype II											
Prototype III											
Préparation pour la présentation finale											

Les  représentent les rencontre du client 1, 2 et 3 ainsi que la journée de la conception

4. Processus de conception



a. Enoncer du probleme

"Services Partager Canada cherchent une solution qui vas répondre au besoin de la surcharge informatique pour leur employé. Cette solution qui serait terminer dans le cadre des trois prochains mois, peut avoir des propriétés qui font sure que les employés reçoivent et lisent les messages importants envoyer par la compagnie sans déranger leur travaille."





TABLEAU DE SPECIFICATION

Exigence fonctionnelle

Exigences fonctionnelles					
Besoins du client	Critère de conception	Relation	Valeur	Unités	Méthode de vérification
Crée une plateforme ou un outil qui catégorisera les informations selon leur importance et leur urgence	Catégorisation des messages, Nombre de courriels traités par minute	≥	1000	Courriels/minutes	Test de charge
La plateforme pourrait faire des mises à jour quand les mesges importants changent.	Capacité de stockage, taille maximale de courriels,	≤	2	Mo	Vérification et test d’envoi
Faire un produit qui est compatible avec les logiciels courriel, Outlook et Microsoft.	Compatibilité de la plateforme ou de l’outil	≥	96	%	Test de compatibilité
Les messages seront contrôlés par un outil qui enverra la notification ou afficherais un message quand l'information a été lus par les employer.	Suivi de lecture des courrielle, délai d’envoi des courriels	≤	5	Minute	Tests unitaires
Le produit va être tester par une variété de personnes afin qu’il so it bien adapté pour cet environ nement et ce groupe cible	Test d’utilisateur, Temps de réponse du système	≤	2	Seconde	Test de performance
L’outil qui ne laissera pas les messages passer inaperçu par les employés	Système de notifications, Son a l’envoi du courriel	=	Oui	/	Test
Le produit pourrait faire une notification forte ou même avoir une petite vibration.	Système de notification, Vibration pour alerte	=	Oui	/	Test



EXIGENCES NON-FONCTIONNELLES

Exigences non- fonctionnelles					
Besoins du client	Critère de conception	Relation	Valeur	Unités	Méthode de vérification
Le système pourra alerter les employées si certaines pratiques environnementales ne seront pas respectés et le système a un petit impacte environnementale	Impact environnemental	=	Oui	/	Test
Crée un système personnalisable que les employées pourront changer selon leurs préférences	Personnalisation	=	Oui	/	Test
Si notre produit est une solution physique on peut prendre avantages de ces places public et souvent fréquenté pour afficher les messages	Affichage public	≥	3	Affichage/ minutes	Test

CONTRAINTES

Besoins du client	Critère de conception	Relation	Valeur	Unités	Méthode de vérification
L'outil ou la plateforme sera facile à installer sur les ordinateurs et téléphone portable et facile à utiliser.	Usage du téléphone, ordinateur, tablette. Facilité d'utilisation	=	Oui	/	Test de performance sur le téléchargement
L'outil ou la plateforme peut envoyer des notifications sans être trop fort ou trop perturbant pour les employés de façon qui ne les distraites pas de leur travaille car leur confort est important	Système de notifications, Pas perturber les employées	=	Oui	/	Test
L'outil rappellera aux employées des lres leurs messages importants ou aurait des notifications captivant pour faire sure que les employer les voient.	Système de notifications, Confirmation de lecture	=	Oui	/	Test
Les notifications pourraient avoir un aspect visuel qui est captivant pour les employés	Système de notifications, Animation dans le message	=	Oui	/	Test

CONCEPT DETAILLER

Pseudocode for handling the message processing and notifications

```
# Import necessary libraries
import os
from notification_system import send_notification, attach_file

# Define function to process the message and its recipients
def process_message(message, priority, recipient_type, specific_people=None, group=None, file=None):

    # Determine recipients based on recipient_type
    if recipient_type == "everyone":
        recipients = get_all_recipients() # Function to fetch all users in the system
    elif recipient_type == "specific":
```

Envoyer un Message

Titre :

Message :

Niveau de priorité :

Icône (image) :

 Aucun fichier n'a été sélectionné

Joindre un fichier (facultatif) :

 Aucun fichier n'a été sélectionné

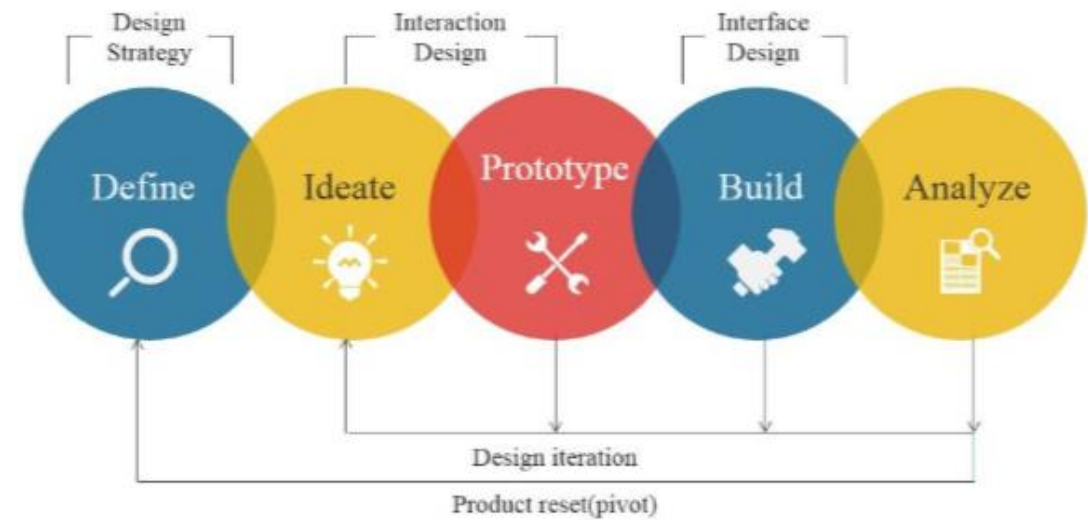
Activer le son :

☐

Autoriser la réponse :

☐

Prototype Development



PROTOTYPE I

Notre premier prototype a été effectué le 27 octobre de cette année, il est plutôt basique et il répond pleinement aux besoins interprétés de nos clients suite à la rencontre 1. Grâce à ce prototype cible analytique, un serveur peut envoyer des messages de degrés et de priorités différentes avec la possibilité d'inclure des images et d'autoriser une réponse potentielle de la part du client.

A) Envoi du message:

- Comme vous pouvez le voir, notre serveur dispose d'une interface qui lui permet d'envoyer les emails ou les messages souhaites.

Envoyer un Message

Titre :

Bonjour

Message :

Comment allez vous cr7 |

Niveau de priorité :

Haute

Icône (image) :

Choisir un fichier Aucun fichier choisi

Joindre un fichier (facultatif) :

Choisir un fichier Aucun fichier choisi

Activer le son :

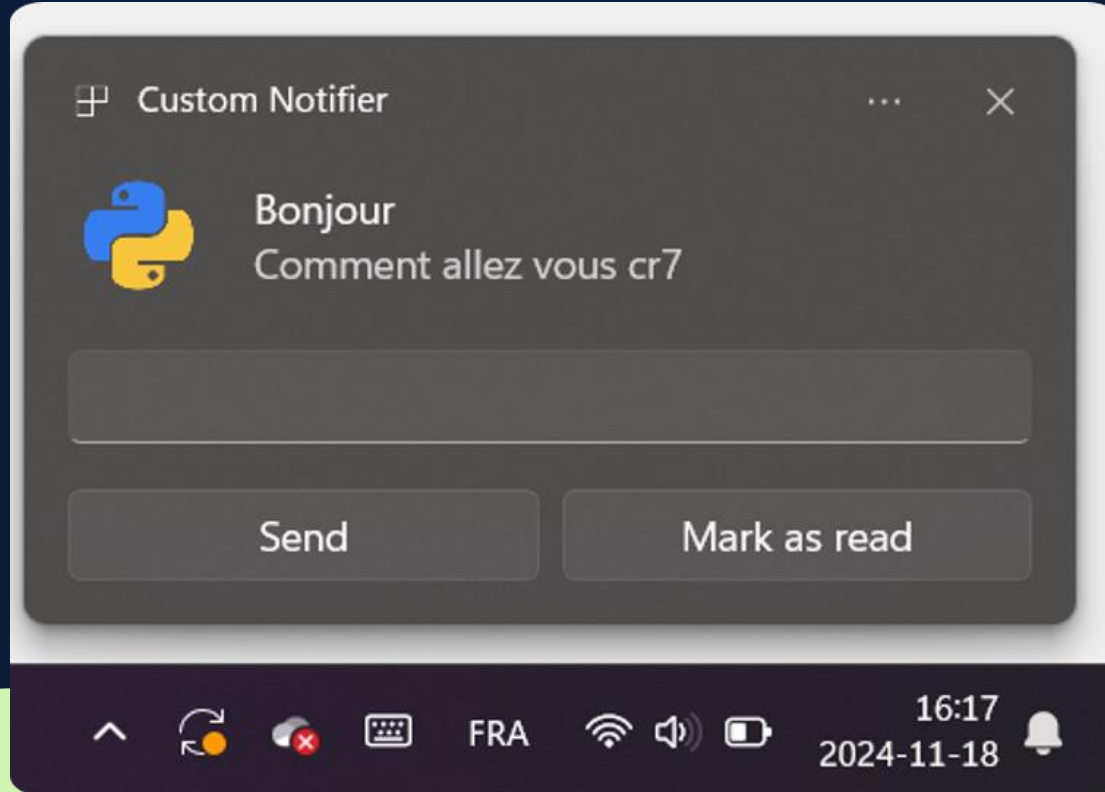


Autoriser la réponse :



Envoyer

B) Réception du message:

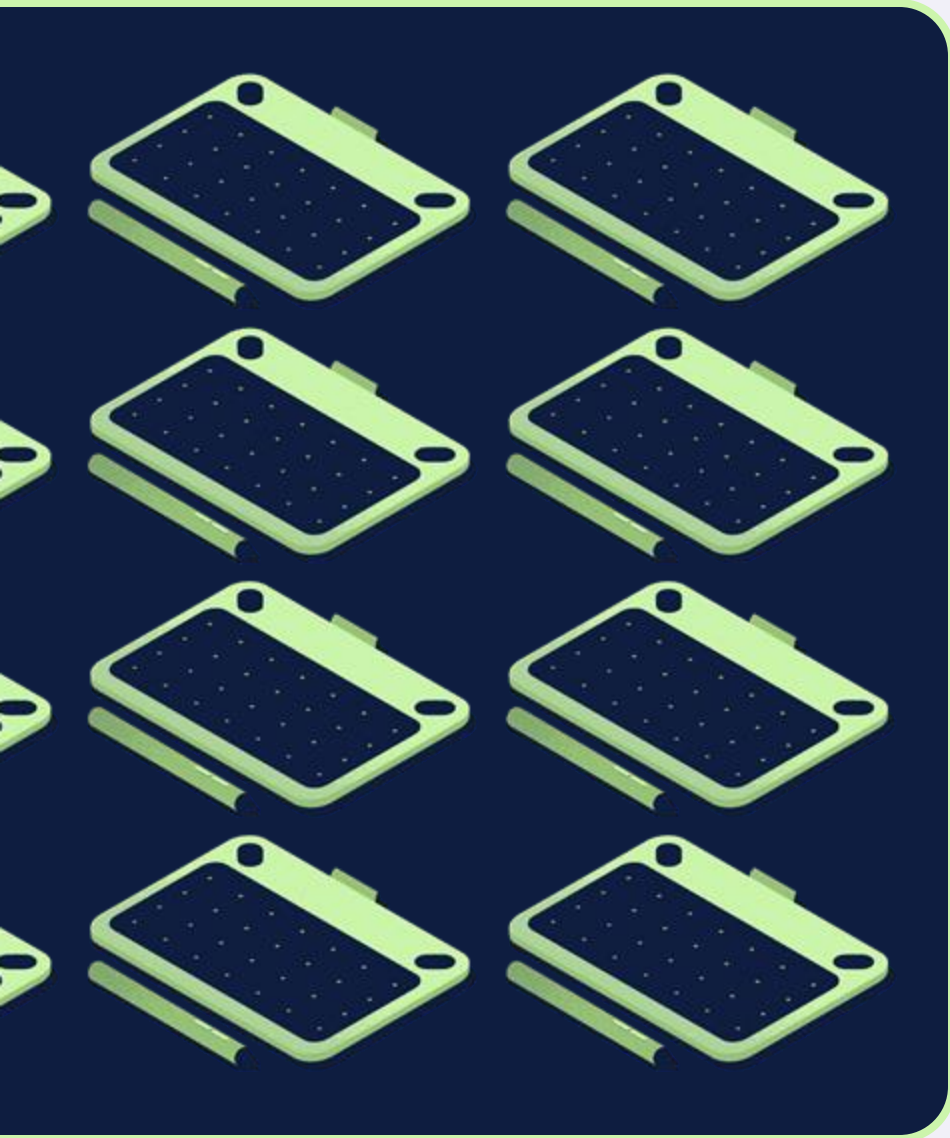


- Une fois le message envoyé, le client reçoit ceci grâce à une notification push qui apparaît en bas de l'écran et qui fait un bruit pour attirer l'attention

C) Réponse au message:

```
C:\Users\amedj\AppData\Loc: × + ▾
Client removed from active connections list: <websockets.legacy.server.WebSocketServerProtocol object at 0x00000182CDDDA350>
Received data from client: Client response: ok
Received data from client: Client response: ok
Client connected: ('::1', 61328, 0, 0)
Received data from client: html
Client removed from active connections list: <websockets.legacy.server.WebSocketServerProtocol object at 0x00000182CDDDA9D10>
Client connected: ('::1', 61338, 0, 0)
Received data from client: html
Client removed from active connections list: <websockets.legacy.server.WebSocketServerProtocol object at 0x00000182CDDDA5D0>
Client connected: ('::1', 61342, 0, 0)
Received data from client: html
Client removed from active connections list: <websockets.legacy.server.WebSocketServerProtocol object at 0x00000182CDDDA0D0>
Client connected: ('::1', 61343, 0, 0)
Received data from client: html
Client removed from active connections list: <websockets.legacy.server.WebSocketServerProtocol object at 0x00000182CDDDA990>
Received data from client: Notification marked as read.
Received data from client: Notification marked as read.
Received data from client: Notification marked as read.
Received data from client: Notification marked as read.
Client connected: ('::1', 63148, 0, 0)
Received data from client: html
Client removed from active connections list: <websockets.legacy.server.WebSocketServerProtocol object at 0x00000182CDDDA9590>
Received data from client: Client response: Salut j'ai bien reçu votre message.
Received data from client: Client response: Salut j'ai bien reçu votre message.
```

Puis celui-ci peut répondre au message et la réponse apparaîtra et sera stockée dans une base de données chez le serveur.



D) Ameliorations

Bien que notre premier prototype soit complètement fonctionnel et répond en partie aux attentes des clients il existe des modifications à faire comme le fait que nous pouvons pas insérer une image

PROTOTYPE II

Objectifs

- Tester le système d'authentification
- Tester le système de commande pour les administrateurs
- Tester les systèmes de base de données

Essais

- Vérifier la connexion avec nom d'utilisateur
- Tester l'ajout de nouveaux utilisateurs par les administrateurs
- Valider la connexion pour tous les utilisateurs et administrateurs



C:\Users\henry\PycharmProjects\pythonPrototype\.venv\Scripts\python.exe C:\Us

Attempting to connect to the WebSocket server...

Connected to WebSocket server.

Enter username: bill

Enter password: 1111

Credentials sent to server.

Admin auth successful.

list of commands:

- add <username> <password> <adminBool>
- rm <username>

Listening for notifications from the server...

Enter a command: rm dan

Enter a command: The following accounts were removed dan

Run



websocketServer x



websocketClient x



websocketClient x



websocketClient (2) x



C:\Users\henry\PycharmProjects\pythonPrototype\.venv\Scripts\python.exe C:\Users\henry\

Attempting to connect to the WebSocket server...

Connected to WebSocket server.

Enter username: admin123

Enter password: admin123

Credentials sent to server.

Admin auth successful.

list of commands:

- add <username> <password> <adminBool>
- rm <username>

Listening for notifications from the server...

Enter a command: add bill 1111 true dan 2222 false

Enter a command: Added bill as admin dan as user



```
C:\Users\henry\PycharmProjects\pythonPrototype\.venv\Scripts\python.exe C:\Users\henry\Py
Attempting to connect to the WebSocket server...
Connected to WebSocket server.
Enter username: user1
Enter password: 1234
Credentials sent to server.
Authentication successful.
Listening for notifications from the server...
Enter a message:
```


Run



websocketServer x



websocketClient x



websocketClient x



websocketClient (2) x



C:\Users\henry\PycharmProjects\pythonPrototype\.venv\Scripts\python.exe C:\Users\henr

Server started

HTTP server running on <http://localhost:8080>

Enter notification title: [data]: {'username': 'admin123', 'password': 'admin123'}

('::1', 59150, 0, 0) authenticated successfully as admin

Client connected: ('::1', 59150, 0, 0)

[data]: {'username': 'bill', 'password': '1111'}

('::1', 59153, 0, 0) authenticated successfully as admin

Client connected: ('::1', 59153, 0, 0)

[data]: {'username': 'dan', 'password': '2222'}

('::1', 59160, 0, 0) failed authentication. incorrect username

[data]: {'username': 'user1', 'password': '1234'}

('::1', 59163, 0, 0) authenticated successfully as client

Client connected: ('::1', 59163, 0, 0)

Améliorations:

Send Notification

Username:

Password:

Notification Title:

Notification Message:

Urgency Level:

Normal

Play Sound:

No

Require Response:

No

Icon File (optional):

Choisir un fichier

 Aucun fichier n'a été sélectionné

Attachment File (optional):

Choisir un fichier

 Aucun fichier n'a été sélectionné

Prototype III

L'ensemble de tous les prototypes



Se connecter

Nom d'utilisateur :

Mot de passe :

Se connecter

Envoyer un Message

Titre :

Saisissez le titre de la notification

Message :

Saisissez votre message ici...

Niveau de priorité :

Haute

Envoyer à :

Groupe

Individus

Sélectionnez un groupe :

Développeurs

Développeurs

Marketing

RH

Gestion des Comptes

Ajouter un compte

Envoyer un Message

Titre :

Saisissez le titre de la notification

Message :

Saisissez votre message ici...

Niveau de priorité :

Haute

Envoyer à :

Groupe

Individus

Supprimer un compte

Sélectionnez un compte :

Henry

Henry

Mico

Amedj

Sophi

Annuler

Envoyer

Gestion des Comptes

Ajouter un compte

Supprimer un compte

Envoyer un Message

Titre :

Saisissez le titre de la notification

Message :

Saisissez votre message ici...

Niveau de priorité :

Haute

Envoyer à :

Groupe

Individus

Envoyer

Gestion des Comptes

Ajouter un compte

Supprimer un compte

Envoyer un Message

Titre :

Saisissez le titre de la notification

Message :

Saisissez votre message ici...

Niveau de priorité :

Haute

Envoyer à :

Groupe

Individus

Sélectionnez des individus :

Henry

Mico

Amedj

Sophi

Envoyer

Gestion des Comptes

Envoyer un Message

Titre :

Saisissez le titre de la notification

Message :

Saisissez votre message ici...

Niveau de priorité :

Haute

Envoyer à :

Groupe

Individus

Ajouter un compte

Nom :

Entrez le nom

Mot de passe :

Entrez le mot de passe

UUID :

Entrez l'UUID

Rôle :

Admin

Ajouter

Annuler

Gestion des Comptes

Ajouter un compte

Supprimer un compte

Project

include

Lib

Scripts

.gitignore

pyvenv.cfg

ai help

ai_help_ai help_we

ai_help_ai help_we

prototype 1

notification_interfa

received_icon.png

websocketClient.p

websocketServer.

computerNotification

login.html

main.html

notification_interface.

output_image.png

pair.py

received_attachment

received_icon.png

RED.png

run_me.py

server_cert.pem

server_key.pem

setup.py

test1.html

test1.py

websocketClient - Copy.py

prototype 1\websocketClient.py

prototype 1\websocketServer.py

login.html

main.html

web

```
65 async def listen_notifications(websocket): 1 usage
99 )
100
101     await asyncio.sleep(0.1)
102
103 async def async_input_loop(websocket): 1 usage
104     while True:
105         user_input = await asyncio.to_thread(input)
106         await to_server(user_input, websocket)
107
108 async def main():
109     uri = "ws://localhost:8765"
110     print("Attempting to connect to the WebSocket server...")
111     try:
112         async with websockets.connect(uri, open_timeout=10) as websocket:
113             print("Connected to WebSocket server.")
114             await websocket.send("Client connected")
115
116             # Start listening to notifications and handling user input concurrently
117             await asyncio.gather(
118                 listen_notifications(websocket),
119                 async_input_loop(websocket)
120             )
121
122     except Exception as e:
123         print(f"Error: {e}")
```

Run

websocketClient

websocketServer

Processing data: 100%| 1.33M/1.33M [00:00<00:00, 12.6MB/s]

Processing data: 100%| 2.05k/2.05k [00:00<?, ?B/s]

Enter notification title:

pythonPrototype


prototype 1

websocketClient.py

108:18 (4422 chars, 125 line breaks)

CRLF

Custom Notifier



test

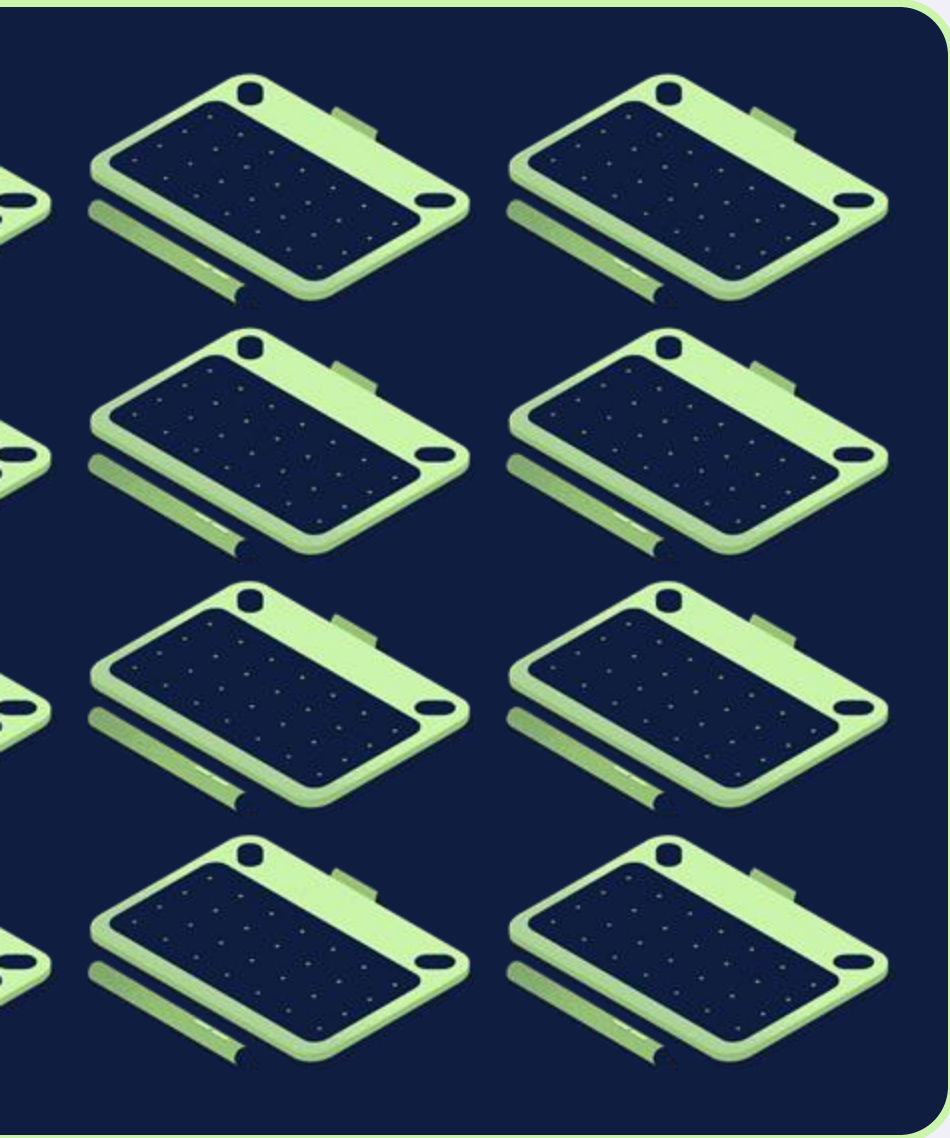
This is a test of the notification system

Send

Mark as read

CONCLUSION

Pour conclure, DASH-N est une solution novatrice et performante pour faciliter la communication en entreprise, en diffusant des informations cruciales sans encombrer les employés. En raison de sa technologie de notifications intelligentes et de sa classification par ordre d'importance, l'application garantit que chaque message parvienne à son destinataire sans interrompre les tâches quotidiennes des employés. Les messages importants sont rapidement transmis assurant donc une communication prioritaire et rapide. Les travailleurs peuvent lire, répondre et confirmer la lecture des messages, ainsi l'entreprise peut suivre la réception, l'implication et la compréhension des informations. Cette méthode simplifie la gestion de la communication interne tout en suivant le processus de travail individuel de chaque employé. En supprimant les distractions superflues et en favorisant une interaction rapide et ciblée ; DASH-N : notre application améliore l'efficacité et renforce la réactivité de l'entreprise en cas de défis de communication importants. De cette manière, grâce à cette solution, l'entreprise garantit que les données clés sont transmises, reçues, lues, comprises et gérées de manière efficace, sans causer de charge mentale excessive ou des interruptions. Ceci encourage une meilleure coordination entre les équipes, ce qui aide à améliorer la performance globale.





LEÇONS APPRISE

- Gestion du temps
- Travail en équipe
- Gestion du budget
- Gestion de conflit

Questions?