

# Rapport de TP : Hébergement du site

## Sommaire :

<b>I. Introduction</b>	<b>1</b>
<b>II. Le travail à réaliser</b>	<b>1</b>
<b>III. Conclusion</b>	<b>5</b>

## I. Introduction

Dans le cadre de mon apprentissage sur l'hébergement de services réseaux, j'ai réalisé ce travail pratique portant sur la mise en place d'un serveur web. L'objectif principal était de comprendre le fonctionnement d'un service HTTP, d'abord à travers une solution ultra-légère comme **TinyWeb**, puis en évoluant vers des environnements de développement plus complexes et professionnels. Ce TP m'a permis d'aborder des notions essentielles telles que la gestion des répertoires racines (DocumentRoot), l'utilisation de l'invite de commande pour le lancement de services, et la virtualisation d'un environnement serveur complet.

## II. Le travail à réaliser

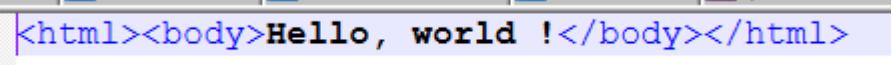
### 1) Installation de TinyWeb

J'ai commencé par télécharger l'archive de TinyWeb. Comme il s'agit d'un serveur ultra-léger et portable, je n'ai eu aucune installation complexe à réaliser : j'ai simplement extrait le fichier exécutable **tiny.exe** dans un dossier dédié sur mon disque C:.



### 2) & 3) Crédit et sauvegarde de la page index.html

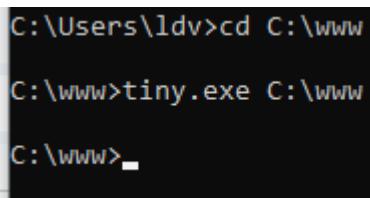
J'ai créé un fichier HTML de base nommé index.html où j'ai collé le code : <html><body>Hello, world !</body></html>. Pour respecter la structure demandée, j'ai organisé mon arborescence en créant un dossier C:\www où j'ai sauvegardé cette page, qui servira de racine à mon site web.



```
<html><body>Hello, world !</body></html>
```

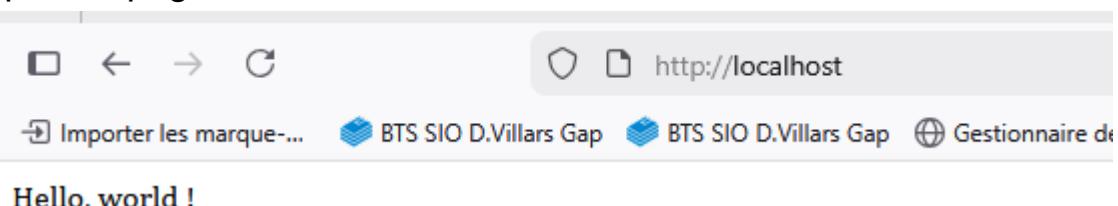
#### 4) Test de visualisation

Lors de ma première tentative de lancement via l'invite de commande, j'ai rencontré une erreur car le système ne trouvait pas le programme dans mon dossier utilisateur personnel. J'ai résolu ce problème en me déplaçant d'abord dans le répertoire C:\www à l'aide de la commande cd, puis en exécutant tiny.exe C:\www. Le serveur est désormais actif et pointe sur mon dossier de site.



```
C:\Users\ldv>cd C:\www
C:\www>tiny.exe C:\www
C:\www>.
```

En me rendant sur <http://localhost> avec mon navigateur, j'ai pu confirmer que ma page 'Hello World' s'affichait correctement.



J'ai ensuite répété l'opération avec les fichiers du site GSB pour vérifier l'affichage des éléments plus complexes.

#### 5) Comparatif des logiciels d'hébergement local

Logiciel	Avantages	Inconvénients
WampServer	Très complet, gestion	Un peu lourd pour des

	facile de MySQL/PHP sous Windows.	tests simples.
<b>XAMPP</b>	Multiplateforme (Windows, Linux, Mac), très populaire.	Interface parfois austère.
<b>Laragon</b>	Moderne, rapide, gestion isolée des projets.	Moins connu des débutants.
<b>TinyWeb</b>	Ultra-léger, sans installation.	Très limité (pas de PHP/Base de données de base).

J'ai effectué des recherches sur les solutions de simulation d'hébergement local (piles WAMP). Après comparaison, j'ai retenu que si TinyWeb est idéal pour du HTML statique, des solutions comme WampServer sont préférables pour des sites dynamiques.

## 6) Choix du logiciel

Suite à l'analyse de la demande des développeurs, j'ai porté mon choix sur **WampServer**. Ce logiciel permet de simuler un environnement serveur complet (Apache, PHP, MySQL) indispensable pour faire fonctionner les scripts du site GSB, ce que TinyWeb ne permet pas.

## 7) Mise en place sur Machine Virtuelle (VM)

Afin d'isoler l'environnement de production, j'ai configuré une machine virtuelle. J'ai veillé à installer les runtimes Visual C++ au préalable, car ils sont indispensables au bon fonctionnement d'Apache sous Windows, puis j'ai procédé à l'installation du serveur avec WampServer

## 8) Rendre opérationnel le site GSB

Pour mettre en ligne le site vitrine GSB, j'ai accédé au répertoire racine de mon serveur WampServer situé dans **C:\wamp64\www\**. J'y ai transféré l'intégralité des sources du site GSB. En remplaçant la page d'accueil par défaut de Wamp par l'index du site vitrine, j'ai pu confirmer

le bon fonctionnement de l'hébergement en visualisant le résultat final via l'adresse <http://localhost>

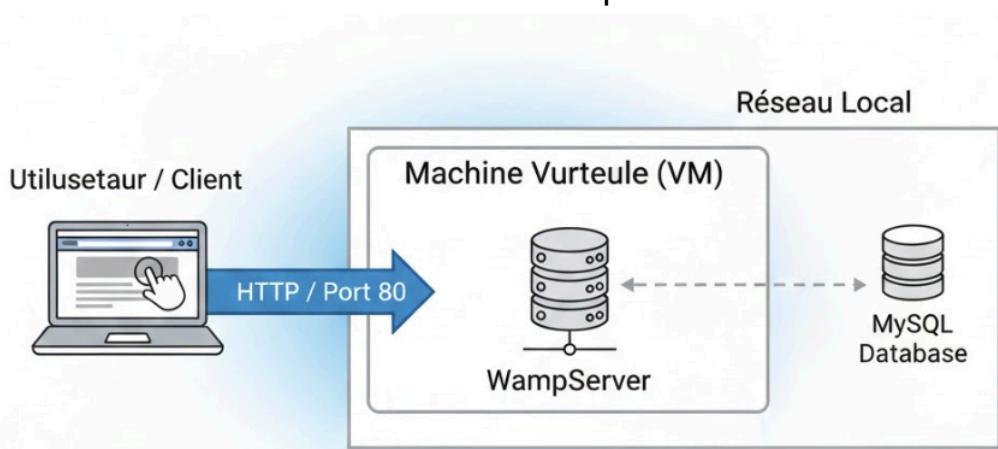
## 9) Moyens d'hébergement externe

Pour simuler ou réaliser un hébergement hors du poste local, j'ai identifié plusieurs solutions allant de l'hébergement mutualisé, plus abordable, au VPS qui offre une liberté totale de configuration.

Moyen	Description
<b>Hébergement Mutualisé</b>	Partage d'un serveur avec d'autres sites (ex: OVH, Hostinger).
<b>VPS (Serveur Privé Virtuel)</b>	Serveur virtuel dédié dont on a le contrôle total.
<b>Cloud (AWS / Azure)</b>	Ressources à la demande et scalables.

## 10) Schéma Réseau

Pour conclure, j'ai schématisé l'infrastructure réseau mettant en évidence l'interaction entre le client (mon navigateur) et le serveur web hébergé sur la machine virtuelle à travers le protocole HTTP.



### III. Conclusion

Ce TP m'a permis de prendre en main les étapes fondamentales du déploiement d'un site web. J'ai commencé par manipuler **TinyWeb** en créant une arborescence dédiée sur mon disque **C:** et en y plaçant mes fichiers sources. Bien que j'aie rencontré une erreur initiale de chemin dans le terminal, j'ai su diagnostiquer le problème et adapter mes commandes pour lancer le serveur avec succès.

Par la suite, l'étude comparative des logiciels comme **WampServer** ou **Laragon** m'a fait comprendre les limites des serveurs portables face aux besoins de sites dynamiques comme celui de **GSB**. La mise en place finale sur une machine virtuelle dédiée m'a offert une vision concrète de l'isolement des services en entreprise. Ce travail constitue une base solide pour mon portfolio, illustrant ma capacité à configurer un environnement d'hébergement local et à documenter des procédures techniques.