



PJS4 // BTTP

L'interopérabilité simple et sécurisée.

PROFESSEUR RÉFÉRENT

Karim FOUGHALI

PRÉSENTÉ PAR

Sara ZAIBET

Luca RANDAZZO

Clément MAUPERON

Robin LANDAIS

Clément PROST



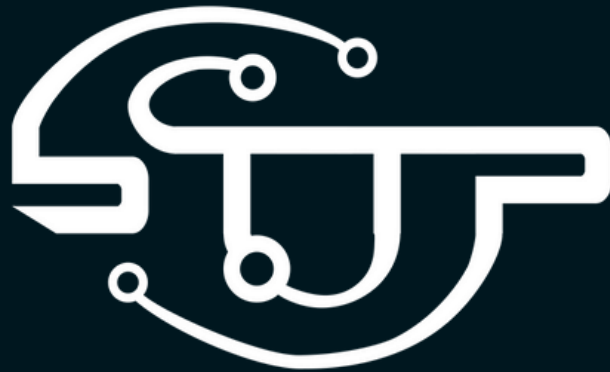
IUT de Paris - Rives de Seine
Université Paris Cité

SOMMAIRE

1. INTRODUCTION	4
2. CONTEXTE	5
2.1 PROBLEMATIQUES	5
2.1.1 INTEROPERABILITE	5
2.1.2 ACCESSIBILITE	5
2.2 SOLUTIONS DÉJÀ EXISTANTES	6
2.3 OBJECTIFS DE REPONSE	7
2.3.1 COMPATIBILITE	7
2.3.2 SECURITE	7
2.3.3 SIMPLICITE	8
3. LE PROJET BTTP	9
3.1 LE PROTOCOLE	9
3.2 LE SERVICE	9
3.3 L'INFRASTRUCTURE	10
4. SOLUTIONS TECHNIQUES	11
4.1 BASE DE DONNEES	11
4.2 APPLICATIONS RESEAUX	12
4.2.1 PROTOCOLE	12
4.2.2 CLIENT	12
4.2.2.1 BIBLIOTHEQUE	12
4.2.2.2 INTERFACE EN LIGNE DE COMMANDE	12
4.2.3 SERVEUR	13
4.3 PLATEFORME WEB	13
4.4 APPLICATION MOBILE	14

4.4.1 DESIGN	
4.4.2 FONCTIONNALITÉS	14
5. RÉALISATION DU PROJET	15
5.1 PRÉPARATION	16
5.1.1 ÉTUDE DE LA VIABILITÉ	16
5.1.2 MESURES DES RISQUES	16
5.1.3 CHARTE GRAPHIQUE	16
5.2 ORGANISATION	17-18
5.2.1 ARCHITECTURE DU PROJET	19
5.2.2 RÉPARTITION DES TÂCHES	19
5.2.3 CHOIX DES TECHNOLOGIES	20
5.2.3.1 PROTOCOLE	21
5.2.3.2 PLATEFORME WEB	21
5.2.3.3 APPLICATION MOBILE	22
5.3 DÉVELOPPEMENT	23
5.3.1 BASE DE DONNÉES	24
5.3.2 APPLICATIONS RÉSEAUX	24
5.3.2.1 PROTOCOLE	25
5.3.2.2 CLIENT	25
5.3.2.3 SERVEUR	26
5.3.3 PLATEFORME WEB	27
5.3.4 APPLICATION MOBILE	28
5.4 MISE EN PRODUCTION	29
5.4.1 LOCATION D'UN VPS	30
5.4.2 CONFIGURATION DOCKER	30
5.4.3 INTÉGRATION DE LA BASE DE DONÉES	31
5.4.3.1 ADMINISTRATION	32
5.4.3.2 INTÉGRATION DES SCRIPTS	32
5.4.3.3 CONNEXION AVEC LES APPLICATIONS	32
5.5 FINALISATION	32
5.5.1 TESTS	33
	33

5.5.2 SCRIPTS DE DÉMONSTRATION	34
6. BILAN	35
6.1 RÉUSSITES	35
6.2 DIFFICULTÉS	36
6.3 AMÉLIORATIONS	37
6.3.1 RENFORCEMENT DE L'INFRASTRUCTURE	37
6.3.2 PROFESSIONNALISATION	37
6.3.2.1 TESTS UNITAIRES	37
6.3.2.2 PLANIFICATION DES VERSIONS	37
6.3.3 DÉVELOPPEMENT DE LA COMMUNAUTÉ	38
6.3.3.1 DÉVELOPPEMENT DU MARKETPLACE	39
6.3.3.2 AMÉLIORATIONS DE L'ACCESSIBILITÉ	40
7. CONCLUSION	41
8. REMERCEMENTS	42
9. ANNEXES	43



INTRODUCTION

L'objectif du PJS4 est de développer un projet permettant à chacun d'approfondir ses compétences en informatique, en expression et en gestion de projet de manière autonome. Le choix libre du sujet a permis de réunir des étudiants ayant les mêmes centres d'intérêt autour d'un projet qui les intéresse réellement. C'est de cette manière que notre groupe s'est formé, naturellement. Clément Mauperon nous a proposé un projet qui nous a tout de suite interpellé : un système d'interopérabilité mobile - PC, qui permettrait d'exécuter des actions sur un PC distant à partir d'un smartphone. Nous avons trouvé le projet à la fois ambitieux, complexe et intéressant. Nous nous sommes longuement concertés sur le sujet, par peur de ne pas le finir à temps vu la complexité technique sur laquelle repose le système. Luca Randazzo et Clément Mauperon nous ont convaincus de travailler le sujet, ayant l'opportunité de développer un projet qui nous tient à cœur pour une fois. Nous nous sommes alors lancé le défi de le développer et de le présenter à temps pour la soutenance. C'est ainsi que nous sommes fiers de vous présenter BTTP, le système d'interopérabilité accessible à tous, pour tous.

CONTEXTE

PROBLÉMATIQUES

INTEROPÉRABILITÉ

Dans le monde d'aujourd'hui, les appareils connectés sont plus nombreux que le nombre des personnes sur Terre. Sachant que certaines régions du monde n'ont pas un accès à des appareils connectés, cela signifie que beaucoup de personnes ont accès à plusieurs appareils connectés.

De plus, il existe une demande pour pouvoir gérer son ordinateur à distance, les téléphones étant au fond un ordinateur mobile.

Ainsi, nous voulons inventer cette interopérabilité, simplifier les échanges entre mobile et ordinateur, dans le but de permettre à une personne d'avoir accès à son périphérique préféré.

ACCESSIBILITÉ

Le fait que le projet soit compatible avec un grand nombre d'appareils était très important pour nous. C'est cela qui a influencé le choix de nos technologies. C'est ce qui nous a poussées à utiliser le langage C++ qui est très largement répandu à la fois sur les ordinateurs et les différents OS, mais aussi sur les objets connectés. Nous avons aussi choisi Flutter pour créer l'application mobile afin qu'elle soit à la fois utilisable sur les téléphones Apple et Android, de plus Flutter permet l'utilisation du C++ ce qui était fondamental pour permettre la liaison entre l'application et le serveur. Nos choix nous permettent donc d'utiliser l'application sur tout type de smartphone et d'utiliser comme terminal tous les objets comprenant le C++ et ayant un accès à un internet.

CONTEXTE

SOLUTIONS DÉJÀ EXISTANTES

Il existe déjà des solutions clés en main qui proposent plus ou moins le même système que BTTP. On peut citer TeamViewer par exemple, qui est un logiciel de contrôle d'accès de terminaux à distance. En effet, il se base sur le partage d'écran du PC distant, mais ne permet pas d'exécuter des actions sur celui-ci. De plus, le système est gourmand sur le réseau dans la mesure où le partage d'écran est lourd à partager et demande ainsi plus de temps d'exécution, ce qui provoque des bugs. Il existe également Parsec, une application de capture de bureau distant essentiellement utilisée pour les jeux vidéo. Pour les mêmes raisons que TeamViewer, l'application est lourde à faire tourner et peut parfois présenter des bugs.

Il existe également une version réservée aux développeurs, nommée SSH. C'est à la fois un programme et protocole informatique de communication sécurisée. En effet, de la même façon que BTTP, SSH propose un échange de clés de chiffrement en début de connexion, mais son utilisation est trop complexe pour les non initiés.

L'objectif de BTTP est de développer un équivalent du protocole SSH utilisable par tous et pour tous. Nous avons créé une solution accessible au grand public, tout en gardant la complexité et l'efficacité de SSH, et en apportant des solutions aux problèmes de TeamViewer et Parsec.

CONTEXTE

OBJECTIFS DE RÉPONSE

COMPATIBILITÉ

Le fait que le projet soit compatible avec un grand nombre d'appareils était très important pour nous. C'est cela qui a influencé le choix de nos technologies. C'est ce qui nous a poussées à utiliser le langage C++ qui est très largement répandu à la fois sur les ordinateurs et les différents OS, mais aussi sur les objets connectés. Nous avons aussi choisi Flutter pour créer l'application mobile afin qu'elle soit à la fois utilisable sur les téléphones Apple et Android, de plus Flutter permet l'utilisation du C++ ce qui était fondamental pour permettre la liaison entre l'application et le serveur. Nos choix nous permettent donc d'utiliser l'application sur tout type de smartphone et d'utiliser comme terminal tous les objets comprenant le C++ et ayant un accès à un internet.

ROBIN

SÉCURITÉ

La sécurité sur Internet est également devenu un sujet majeur ces dernière années, et de plus en plus d'internautes s'intéressent de près à cet aspect des services. On peut noter deux points majeurs de la sécurité sur Internet :

- *la sécurité des communications* : un message ne peut et ne doit être lu que par son destinataire ;
- *la sécurité des données* : la collection de données personnelles doit toujours se faire avec le consentement de l'utilisateur.

CLÉMENT M.

CONTEXTE

OBJECTIFS DE RÉPONSE

SIMPLICITÉ

Notre maître mot du projet est la simplicité. En effet, bien que l'aspect technique soit complexe, notre objectif est de rendre notre application accessible au grand public et que tous les utilisateurs y trouvent leurs intérêts. Pour ce faire, nous avons développé deux interfaces graphiques qui leur permettront d'utiliser notre système simplement.

Premièrement, une application mobile permettra à l'utilisateur d'envoyer des demandes d'exécution de scripts à l'ordinateur distant de leur choix, pourvu qu'il soit allumé.. Grâce à un graphisme simple et intuitif, l'utilisateur pourra consulter et télécharger des scripts publics disponibles dans la marketplace, ou encore ajouter des scripts sur son compte personnel.

Nous avons également développé un site web, disponible à l'URL suivante : <http://resrc.club>. Ce dernier nous sert de site vitrine, qui permettra de faire connaître notre projet. Le site web est simple d'utilisation dans la mesure où chaque onglet est facilement accessible grâce à la barre de recherche et que les fonctionnalités développées sont claires. L'utilisateur pourra consulter les scripts de la marketplace, accéder aux appareils enregistrés sur son compte ou encore en apprendre plus sur notre équipe et nos convictions dans la page d'informations.

LE PROJET BTTP

BTTP, c'est à la fois un protocole réseau et un service mettant à disposition de n'importe qui ce protocole. Nous avons souhaité construire une solution complète afin de prouver la viabilité de notre projet. En effet, le développement du protocole n'aurait pas forcément convaincu à lui seul les non-développeurs de l'utilité de cette technologie.

LE PROTOCOLE

Un des objectifs du projet BTTP est de créer une technologie potentiellement utilisable par n'importe qui et qu'une entreprise pourrait par exemple intégrer à une de ses solutions internes ou commerciales. Nous avons donc choisi de segmenter au maximum notre solution, avec la mise à disposition d'une bibliothèque embarquant le protocole, et d'une autre bibliothèque intégrant les fonctions utiles au client et dépendant de la première bibliothèque. Les développeurs pourront ainsi utiliser librement nos solutions selon leurs besoins.

La technologie BTTP intègre également la sécurité à sa racine. Ainsi, le protocole garantit le chiffrement des communications et l'authentification des appareils grâce au standard PGP (*Pretty Good Privacy*). L'intégration du chiffrement dans la technologie en elle-même nous paraissait le meilleur moyen de garantir la sécurité des échanges et des appareils.

CLÉMENT M.

LE SERVICE

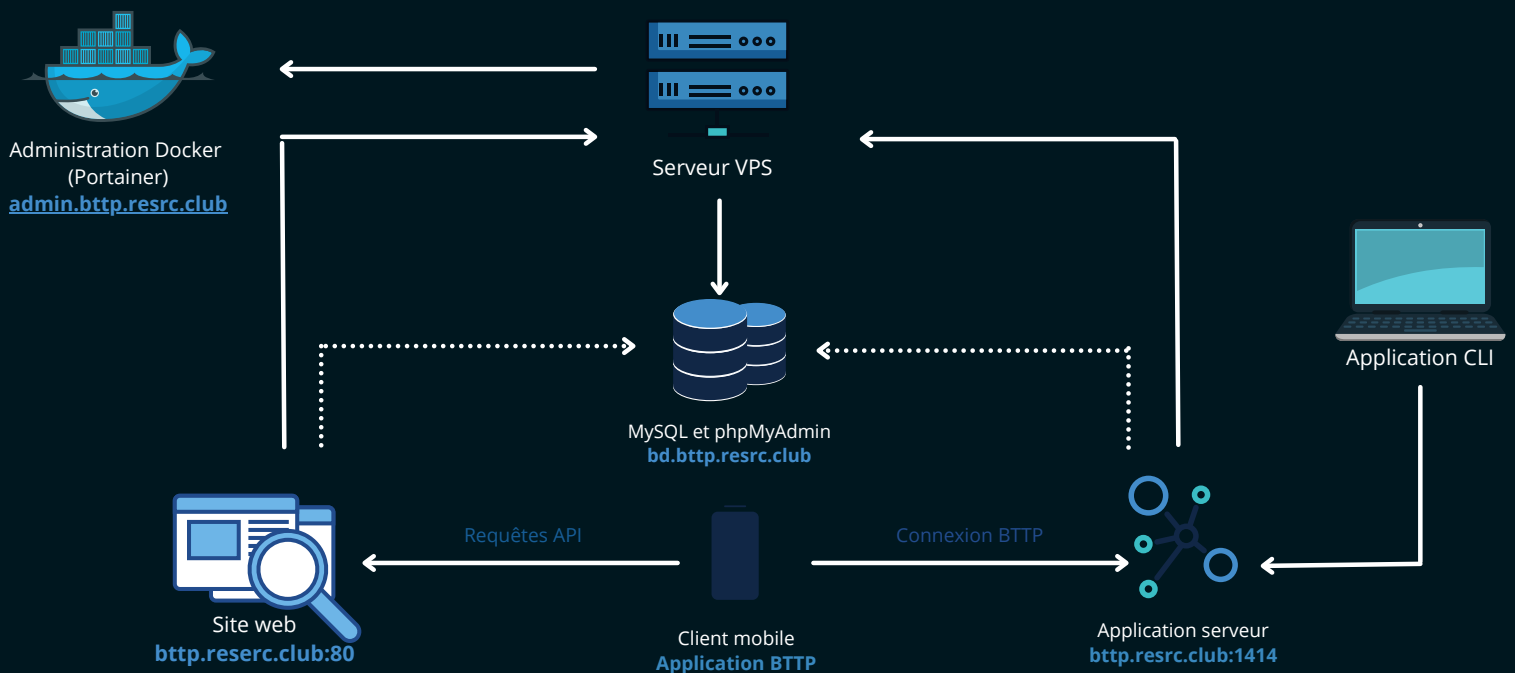
Le service est composé de 2 interfaces et un protocole. Les interfaces sont le site et l'appli mobile et proposent des services aux utilisateurs. Le protocole est le cœur de ce système, étant le véritable produit que nous proposons. L'appli permet de gérer et d'utiliser directement le protocole, tandis que le site permet à un utilisateur d'avoir une expérience plus confortable que sur le client ordinateur disponible, puisque pour l'instant ce dernier n'est que ligne de texte.

CLÉMENT P.

LE PROJET BTTP

L'INFRASTRUCTURE

Le service BTTP nécessite une infrastructure pour fonctionner, bien que celle-ci soit assez simple. BTTP requiert surtout un serveur qui hébergera l'application serveur. Nous y hébergerons aussi le site web et la base de données communes. Docker nous permettra d'administrer ces différentes applications indépendamment les unes des autres (voir [Configuration Docker](#)).



SOLUTIONS TECHNIQUES

BASE DE DONNÉES

La base de données présentement conçue, présente 6 tables :

- *catégorie_script* : définissant les fonctions des scripts disponibles, par exemple : Téléchargement
- *format_script* : définit le format du script, c'est à dire son environnement d'exécution, par exemple : bat, py, sh
- *script_public* : les différents scripts, des informations les concernant (date d'ajout, auteur, etc...) mais également le chemin vers le code source du script
- *script* : liaison entre un *script_public* et un utilisateur
- *utilisateur* : utilisateurs préalablement enregistrés via le site web et permettant l'utilisation de notre système
- *terminal* : la gestion des terminaux ajoutés par les utilisateurs, que ce soit des terminaux clients (smartphones & PC) ou des terminaux distants (PC).

SOLUTIONS TECHNIQUES

APPLICATIONS RÉSEAUX

PROTOCOLE

Afin de garantir la réutilisation future de la technologie, nous avons développé le protocole sous la forme d'une bibliothèque (ensemble de fonctions et d'objets pouvant être intégrés à un projet) facilement réutilisable par des développeurs.

Le protocole intègre directement plusieurs notions de sécurité, telles que la gestion des identités des appareils, le chiffrement et le déchiffrement. Cela permet aux applications BTTP de se baser sur un même "langage de communication" et assure ainsi la cohérence des systèmes. Nous avons choisi d'utiliser le standard PGP (*RFC 4880*) car l'utilisation des signatures de messages PGP et le chiffrement asymétrique sont pour nous le choix idéal puisqu'il permet d'assurer l'intégrité ainsi que la provenance des données. De plus ce standard étant adopté par la communauté informatique notamment via les échanges de mails en utilisant l'outil GPG (*GNU Privacy Guard*).

Le protocole fait abstraction de la connectivité réseau, ce qui permet à chaque développeur d'adapter facilement le protocole à l'utilisation qu'il recherche.

CLIENT

BIBLIOTHÈQUE

La bibliothèque du client dépend de la bibliothèque du protocole et y ajoute la notion d'échanges réseaux. Elle intègre également une gestion des fichiers locaux comme l'identité de l'appareil ou les scripts exécutables sur celui-ci. Elle permet notamment d'exposer l'utilisation de la bibliothèque du protocole aux développeurs souhaitant la réutiliser, et également de fournir une interface simple permettant de communiquer en utilisant le protocole.

INTERFACE EN LIGNE DE COMMANDES

Dans le but de fournir un outil concret permettant d'interagir avec la bibliothèque cliente sur un ordinateur, nous avons également développé une application dite CLI (*Command Line Interface*) nommée *bttp-cli*. Elle est faite pour les ordinateurs Linux uniquement et propose plusieurs commandes pour gérer un client BTTP.

LUCA & CLÉMENT M.

SOLUTIONS TECHNIQUES

APPLICATIONS RÉSEAUX

SERVEUR

Pour que notre infrastructure fonctionne, il nous a fallu développer une application serveur qui évoluera sur notre serveur en parallèle du site web. L'application utilise la bibliothèque cliente évoquée plus haut pour communiquer avec les clients.

Cette application est en réalité un des points centraux de notre infrastructure puisqu'elle est chargée d'interconnecter les différents clients et fait le lien entre la base de données et le service réseau BTTP.

CLÉMENT M.

PLATEFORME WEB

La plateforme web est la vitrine de ce projet, mais permet aussi d'accéder à certaines fonctionnalités facilement. La raison pour laquelle nous avons choisi de faire un site web pour gagner en visibilité, nous pensons qu'une application mobile et un protocole uniquement ne permettraient pas de gagner suffisamment de traction et de visibilité.

Sur le site web, il est possible d'accéder à la marketplace ainsi qu'aux produits et la page d'information en tant que visiteurs. Il est aussi possible de se connecter afin d'accéder aux pages abonnées, qui permettent de gérer son profil, ses appareils ainsi qu'à la marketplace.

Comme on peut le voir, le site web ne propose pas directement les services liés au protocole mais permet de montrer aux visiteurs les possibilités liés au protocole ainsi que d'offrir aux abonnés un endroit facile d'accès ou gérer leur profil et appareils.

SARA & CLÉMENT P

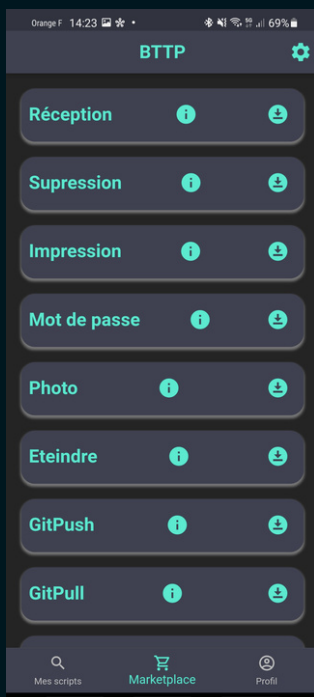
SOLUTIONS TECHNIQUES

APPLICATION MOBILE

DESIGN

L'application mobile est le client de notre projet. Elle est très importante puisque c'est ce que verront les utilisateurs. C'est pourquoi il était très important pour nous qu'elle soit facilement utilisable par tous. L'application était le meilleur choix possible puisque de nos jours, tout le monde possède un smartphone et serait capable d'utiliser notre outil.

Ainsi le plus important pour était de rendre l'utilisation facile et agréable pour les utilisateurs. Pour ce qui est de rendre l'utilisation simple, nous pensons que l'important était de limiter le texte et les actions de l'utilisateur.



Par exemple, ici, nous avons la page principale. Elle est composée de plusieurs cartes qui sont les scripts que l'application peut lancer. On peut voir le nom du script, sur quel terminal il est installé et on peut le lancer grâce à une simple flèche. La barre de navigation en bas de l'application est très simple, il n'y a que 3 pages disponibles qui sont représentées par leurs noms et un logo permettant de comprendre leurs utilités. Nous avons donc décidé de limiter les fonctionnalités pour nous concentrer sur l'essentiel de l'application et la rendre donc très simple d'utilisation.

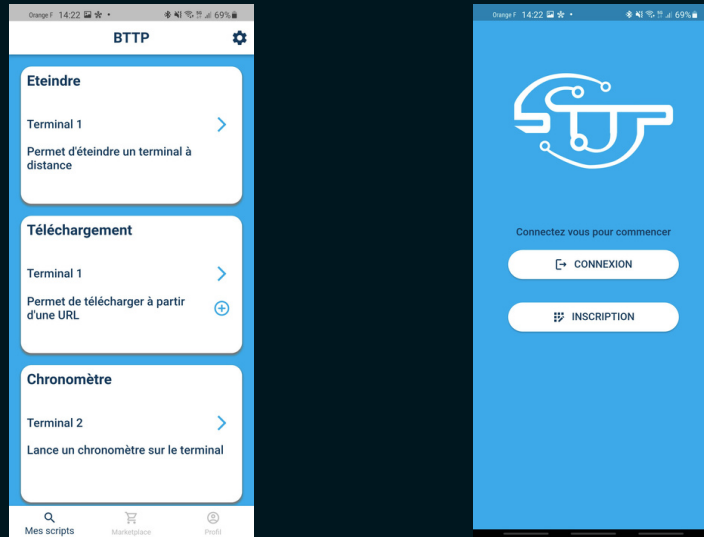
Pour ce qui est de rendre l'application plus agréable, nous avons décidé de développer un thème sombre. Ce dernier est très répandu sur les applications mobiles et permet à l'utilisateur un meilleur confort visuel en abaissant la luminosité et en modifiant les couleurs pour les rendre plus sombres.

On utilise donc des couleurs sombres comme du noir et différents tons de gris. La couleur du texte ressort bien sur un fond sombre sans être trop vive ce qui n'aurait pas été agréable pour l'utilisateur.

ROBIN

SOLUTIONS TECHNIQUES

APPLICATION MOBILE DESIGN



FONCTIONNALITÉS

Le nombre de fonctionnalités dont dispose l'utilisateur est limité. Dans un premier temps, il peut lancer les scripts qui sont à sa disposition. L'utilisateur a aussi accès à un MarketPlace, un endroit où l'utilisateur pourra télécharger des scripts. Ce dernier pourra être amélioré par les futurs utilisateurs qui écriront de nouveaux scripts. Enfin, il y a une page de profil, elle permettra à l'utilisateur de changer de mot de passe et de se déconnecter.

REALISATION DU PROJET

PRÉPARATION *ETUDE DE VIABILITÉ*

Avant de se lancer dans la conception de notre système, il nous a fallu étudier sa viabilité. En effet, notre projet est complexe d'un point de vue technique, c'est pour cette raison qu'il n'existe pas encore de solutions accessible au grand public. Après de nombreuses discussions avec Monsieur Foughali, qui nous a mis en garde sur ce point, nous avons tiré la conclusion que le projet serait en effet intéressant à développer dans le mesure où il pourra être utilisé par tout le monde grâce aux interfaces graphiques que nous développerons, en parallèle du protocole et la partie logique de notre système.

MESURE DES RISQUES

Comme dans chaque projet, il existe des risques liés à la conception. Notre projet se composant de trois produits, le plus gros risque que nous avons encouru est la mise en commun de produits développés en différents langages. En effet, la compatibilité des langages n'est pas toujours assurée, ce qui poserait problème dans l'intégration du protocole à l'application mobile. Nous avons évidemment anticipé ce risque, et nous nous sommes assurés que le langage utilisé pour l'application était bien compatible avec le C++ du protocole.

Le deuxième risque, d'ordre marketing, est que le public ne soit pas séduit par notre projet. Le projet restant un système informatique complexe, il se peut que certaines personnes ne se sentent pas capable de l'utiliser ou n'y voient pas l'utilité. C'est pour cette raison que nous avons développé un site web en plus de l'application, qui offre toutes les réponses aux questions des utilisateurs et qui présente un design attractif et familier.

SARA

REALISATION DU PROJET

PRÉPARATION CHARTE GRAPHIQUE

Notre projet possède une identité graphique réfléchie et justifiée. Nous avons choisi une palette de couleurs essentiellement composée de bleu, qui est une couleur largement répandue dans le domaine de l'informatique. On peut citer quelques exemples :



logo C++



logo GoLang



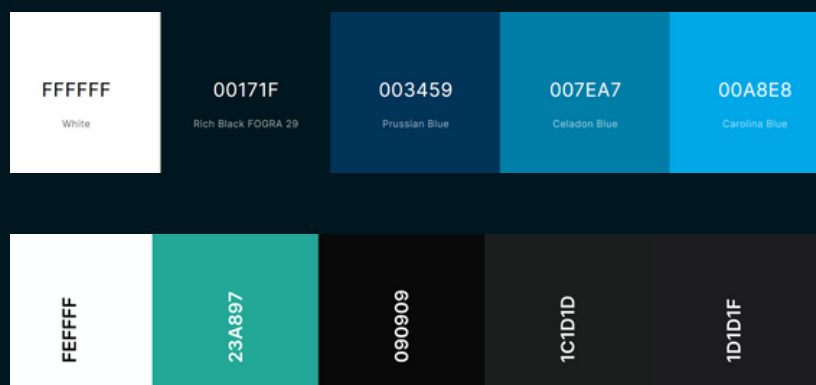
logo CSS



logo IBM

Au-delà de l'aspect informatique, le bleu évoque l'harmonie et le professionnalisme.

Nous avons également pensé à une palette de couleur pour notre thème sombre, qui se compose essentiellement de nuances de noir, et d'un bleu-vert assez vif pour contraster certains éléments graphique, comme les boutons par exemple. Le blanc sert essentiellement au texte, afin qu'il puisse être lisible sur fond noir.



SARA

REALISATION DU PROJET

PRÉPARATION CHARTRE GRAPHIQUE

Pour notre logo, nous avons voulu garder le nom BTTP, acronyme de *Brette Text Transfer Protocol*, en ajoutant un design à l'image de notre projet. Comme celui-ci est essentiellement basé sur l'aspect réseau de l'informatique, nous nous sommes inspirés de deux logos qui évoquent ce même aspect : Apache Kafka et Intel. Nous avons repris les ronds du logo Kafka qui évoquent les nœuds d'échanges et les sous réseaux, auquel nous avons ajouté la forme ronde du logo intel.



logo Intel

+



logo Apache Kafka

=



logo BTTP

Nous avons choisi de nommer notre projet BTTP, en référence à un TP de Java Serveur effectué avec Monsieur Brette. En effet, ce TP nous a inspiré à développer un protocole qui nous est propre, et pour rendre hommage à l'IUT et à Monsieur Brette, nous avons décidé de garder ce nom.

REALISATION DU PROJET

ORGANISATION

ARCHITECTURE DU PROJET

Le projet s'est articulé de la manière suivante :

- Protocole : coeur du projet, essentiel au fonctionnement logique du système. Il s'agit également de la configuration du serveur, du développement du client ainsi que la conception d'un CLI Client
- Application mobile : permet de faire jouer le rôle de client à un smartphone. C'est ce qui permettra de mettre en application le protocole, en envoyant des demandes d'exécution de script sur le terminal distant.
- Site web : site vitrine permettant à un utilisateur abonné de voir ses appareils, peu importe où il se trouve. Pour les visiteurs, ils pourront consulter les scripts publics disponibles dans le marketplace ou encore télécharger l'application client pour PC ou l'application mobile

Rappelons que ces trois produits sont essentiels au déploiement de notre solution. En effet, il s'agit d'un projet complet que nous allons faire évoluer dans un futur proche. Il est donc essentiel d'avoir des interfaces graphiques, tels que le site ou l'application mobile, afin de nous faire connaître et que les utilisateurs non initiés s'y retrouvent.

REALISATION DU PROJET

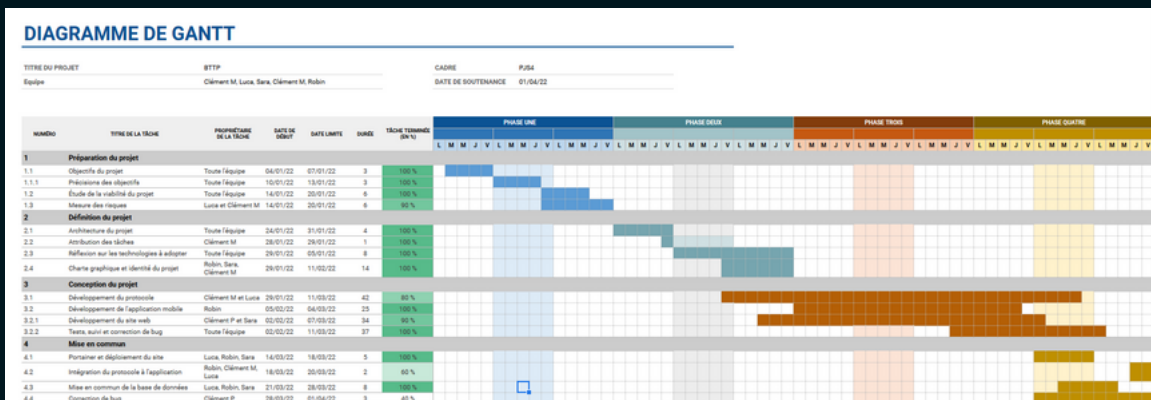
ORGANISATION RÉPARTITION DES TÂCHES

Notre projet étant conséquent et ambitieux, il nous a fallu répartir les tâches de manière équitable et justifiée. Pour cela, nous avons pris avantage des forces de chacun afin d'être efficace dans le développement du système. Clément M. et Luca étant plus à l'aise en C++, ils ont développé le protocole et l'intégration de celui-ci au serveur et au client.

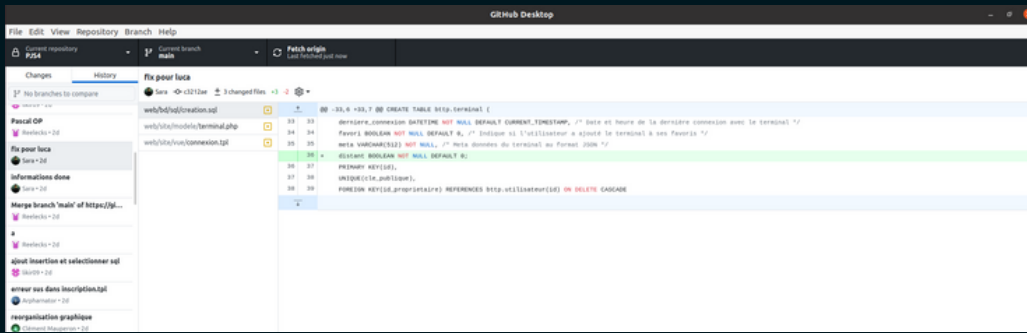
Robin a déjà développé dans le passé des applications mobiles avec Flutter, il a alors développé notre application mobile en utilisant ce framework.

Quant au site web, Sara et Clément P l'ont développé en PHP et Bootstrap dans la mesure où ils sont particulièrement familiers à ces langages, comme ils avaient auparavant développé plusieurs sites web avec ces technologies.

Dans le but de présenter une première version du projet à temps, il nous a fallu développer nos trois produits en parallèle. Ci-dessous, le diagramme de Gantt détaillant nos différentes phases de développement :



Nous avons également mis en place un repos github, dans lequel nous publions nos modifications. De cette manière, nous pouvons suivre les avancements de chacun et revenir sur une version antérieure du code si la nouvelle version ne fonctionne pas. Voici un exemple de suivi sur Github Desktop :



ORGANISATION CHOIX DES TECHNOLOGIES PROTOCOLE

D'abord, nous avons rapidement choisi de développer l'entièreté des bibliothèques (protocole et client) et des applications (CLI et serveur) en C++. En effet, bien que ce langage soit beaucoup moins accessible que le Java par exemple, il est beaucoup plus adapté au développement de technologies bas niveau et propose davantage de compatibilité avec les objets connectés. Aussi, le C++ offre généralement de meilleures performances, ce qui est très important pour une technologie réseau, d'autant plus que les échanges sont chiffrés et déchiffrés.

Ensuite, nous avons utilisé CMake afin de simplifier la compilation et la gestion des dépendances. Cela nous a notamment permis de lier les bibliothèques et les applications, de manière à ce qu'elles soient générées ensemble. Par exemple, lorsqu'on lance la compilation de l'application serveur, les dernières versions des bibliothèques du protocole et du client sont générées également. De la même manière, nous avons pu automatiser la génération des tests pour chaque sous-projet.

Enfin, certaines fonctions des bibliothèques sont déléguées à des bibliothèques tierces. Ainsi, l'implémentation du standard PGP dans le protocole, qui gère notamment le chiffrement des messages, est une bibliothèque C++ sous licence MIT nommée OpenPGP et développée par l'utilisateur de GitHub calccrypto. Nous avons aussi choisi de déléguer la gestion des échanges réseaux à la bibliothèque ASIO, surtout pour son aspect multi-plateformes qui correspondaient à nos besoins.

CLÉMENT M.

REALISATION DU PROJET

ORGANISATION

CHOIX DES TECHNOLOGIES

PLATEFORME WEB

Le site web a été développé en PHP pour le back-end, et en HTML/CSS/Bootstrap pour le front. Le PHP nous a permis de créer des pages web dynamiques et d'automatiser un grand nombre de tâches répétitives sur le site, comme la navigation ou le téléchargement de script public depuis la marketplace. C'est un langage qui possède une grande documentation sur Internet, ainsi nous pouvions la consulter dès que l'on rencontrait un problème. Par ailleurs, PHP est un langage de programmation open source, contrairement à JavaEE par exemple, qui est un langage de programmation essentiellement utilisé en entreprise et qui ne possède pas la diversité de contributions que propose PHP. Ce dernier est donc conforme à notre philosophie de partage et nous avons trouvé cela judicieux de l'utiliser pour développer notre site web.

Le choix de Bootstrap pour le front-end s'est essentiellement basé sur son utilisation courante dans le développement web. En effet, la plupart des sites web sont conçus en Bootstrap. Comme notre objectif est de rendre notre projet compréhensible par le grand public, il est essentiel que notre site vitrine soit familier, facile de compréhension et d'utilisation et que la navigation soit fluide et intuitive. Bootstrap est un excellent framework pour développer des sites pareils dans la mesure où il présente un design familier dans lesquels les utilisateurs non initiés s'y retrouveront facilement.

REALISATION DU PROJET

ORGANISATION

CHOIX DES TECHNOLOGIES

APPLICATION MOBILE

L'application a été développée grâce au framework Flutter et au langage Dart. Il y a plusieurs raisons à ce choix. Le premier est que les applications développées grâce à Flutter sont compatibles avec le C et le C++ ce qui est primordial pour nous. Ensuite, Flutter permet de créer des applications à la fois utilisables sur Android et IOS ce qui permet l'accès à un plus grand nombre. Flutter est une technologie très récente créée par Google qui est rapidement devenue très utilisée dans le développement mobile. Pour créer l'application mobile, nous avons aussi utilisé VSCODE qui permet de compiler le projet afin de tester l'application. Le langage Dart permet aussi de faire facilement des requêtes pour accéder à des API ou à des bases de données. C'est très utile notamment pour accéder aux scripts du MarketPlace ou bien pour la connexion des utilisateurs.

REALISATION DU PROJET

DÉVELOPPEMENT

BASE DE DONNÉES

Pour des raisons de praticité nous avons conçu différents scripts SQL nous permettant de réaliser des tests en local plus rapidement et plus efficacement. Nous avons un script de création relatif à la création de la base et des tables nécessaires. Un script de population dont l'utilité est de remplir ces tables pour nous permettre la réalisation des tests.

REALISATION DU PROJET

DÉVELOPPEMENT APPLICATIONS RÉSEAUX PROTOCOLE

Nous avons commencé à travailler sur le protocole en nous appropriant la bibliothèque *OpenPGP* qui nous servirait à générer des paires de clés PGP, ainsi qu'à chiffrer, déchiffrer et signer des messages à l'aide de ces clés. Nous avons encapsulé l'ensemble de ces fonctions dans un objet appelé Identité. Il représente l'identité d'un appareil, tel qu'un client ou un serveur, lors des échanges BTTP. Certains problèmes sont apparus avec la bibliothèque *OpenPGP*, qui n'est plus en développement actif depuis 2019, et nous avons ajouté une bibliothèque C++ appelée *segvcatch* afin de corriger ces erreurs.

Notre travail s'est ensuite focalisé sur l'implémentation des messages BTTP, qui seraient la base des communications entre les appareils. Chaque message possède un type et un contenu, qui peuvent être sérialisés en un texte, qui sera lui-même chiffré et signé par une identité puis envoyé sur le réseau. Les échanges sont synchrones : un appareil demande une information à un autre appareil et attend de recevoir la réponse avant de continuer.

Lors de la transmission des messages au serveur qui les relayera à l'appareil distant, les messages chiffrés sont encapsulés dans des messages de contrôle qui possèdent eux aussi un type et un contenu (le message chiffré), ainsi qu'une entête qui contient les informations qui doivent être vérifiées par le serveur avant de relayer le message. Par exemple, lors d'une demande d'exécution, l'entête contient l'identifiant du script à exécuter et le serveur vérifie que cet identifiant existe bien dans la base de données. Le serveur de contrôle ne lit donc jamais le contenu du message, qui est de toute façon chiffré. De plus, les messages de contrôle sont sérialisés, puis chiffrés et signés à destination du serveur de contrôle, ce qui sécurise la transmission du message sur le réseau.

Une fois les différents types de messages définis, nous avons retranscrit les comportements des différents appareils (client, contrôle et distant) dans des classes appelées Transactions. Elles traduisent le rôle de chaque appareil dans une transaction BTTP.

CLÉMENT M.

REALISATION DU PROJET

DÉVELOPPEMENT *APPLICATIONS RÉSEAUX* *CLIENT*

Le protocole terminé, nous avons commencé le développement de la bibliothèque cliente qui sera plus tard intégrée à l'application mobile et au serveur. D'abord, nous avons créé un système de gestion des identités de l'appareil, avec la possibilité d'exporter et d'importer des clés privées et publiques depuis des fichiers. Cette étape fut compliquée puisque le standard PGP est très strict sur le format de lecture.

Dans l'objectif d'avancer plus rapidement, nous avons développé en parallèle de cela la structure du client en ligne de commande. Nous avons ainsi pu ajouter des commandes au fur et à mesure de nos avancées sur la bibliothèque, nous permettant ainsi de les tester directement via le CLI.

Nous avons ensuite implémenté les connexions réseaux à l'aide de la bibliothèque ASIO. Cela nous a ensuite permis d'ajouter un gestionnaire de serveurs de contrôle pour que le client choisisse à quel serveur il souhaite se connecter.

Enfin, nous avons ajouté un gestionnaire d'autorisation des exécutions et un gestionnaire des scripts locaux, ce qui nous a finalement permis d'implémenter les deux modes de connexion possible pour un client : en mode client (demande d'exécution) ou en mode distant (réponse à des demandes d'exécution).

CLÉMENT M.

REALISATION DU PROJET

DÉVELOPPEMENT

APPLICATIONS RÉSEAUX

SERVEUR

L'application serveur intègre la bibliothèque client, ce qui nous a grandement facilité le travail. Il ne nous restait plus qu'à créer le système permettant d'accepter les connexions des clients. Pour cela, nous avons créé un objet appelé Session qui s'exécute indépendamment du fil d'exécution principal du serveur et qui répond exclusivement à un client.

Nous avons également dû implémenter une interface de connexion avec la base de données, et ce à l'aide du MySQL C++ Connector. Cela a notamment rendu fonctionnel le système d'authentification.

REALISATION DU PROJET

DÉVELOPPEMENT

PLATEFORME WEB

Nous avons développé le site web suivant le modèle MVC (Modèle Vue Contrôleur). C'est un modèle de conception largement répandu dans le développement web qui consiste à séparer le code en trois parties :

- **Modèle** : c'est dans ce dossier que nous gérons les échanges avec la base de données, correspond au back-end à proprement parlé
- **Vue** : interface graphique qui sera vue par l'utilisateur
- **Contrôleur** : la logique concernant les actions effectuées par l'utilisateur. Fait le lien entre le modèle (la partie logique de données) et la vue (l'interface graphique)

De cette manière, nous avons développé un site web dynamique rapidement. Comme nous étions deux à travailler sur le site, nous nous sommes répartis les tâches dans le but d'être efficaces. Clément P. s'est surtout focalisé sur le back-end, notamment sur le code permettant d'interagir avec la base de données. Il a développé les fonctions du Modèle qui seront utilisées dans le Contrôleur qui fera la liaison avec la Vue. Sara s'est essentiellement concentrée sur le front-end et le design du site. En tirant profit des avantages de Bootstrap, elle a pensé à un design attractif dans lequel les fonctionnalités seront facilement intégrables.

REALISATION DU PROJET

DÉVELOPPEMENT

APPLICATION MOBILE

Pour ce qui est du développement de l'application mobile, nous avons créé des pages dynamiques ce qui permet d'afficher en temps réel les éléments des pages. Nous avons séparé au maximum les différentes pages et méthodes pour simplifier au maximum l'utilisation de notre code (dans notre logique d'Open source). C'est Robin qui a codé toute l'application, il a aussi été aidé par Luca et Clément M. pour l'intégration du C++. Pour la création, nous avons utilisé les dernières fonctionnalités de flutter permettant de créer des objets dédiés pour chaque utilisation requise. Nous avons aussi utilisé plusieurs librairies Flutter pour créer des animations ainsi que le thème sombre.

ROBIN

REALISATION DU PROJET

MISE EN PRODUCTION

LOCATION D'UN VPS

La mise en production de notre service a nécessité la location d'un VPS (*Virtual Private Server*) afin de déployer la base de données, le site web et l'application serveur. Nous avons opté pour l'hébergeur *OVH* car nous possédons déjà un nom de domaine (*resrc.club*) chez eux, cela nous facilite donc l'administration. Notre serveur actuel dispose de la configuration minimale, soit un cœur virtuel, 2 GB de mémoire vive et 40 GB de stockage, ce qui est suffisant pour héberger notre solution. Cela démontre par ailleurs le faible coût de la mise en place d'une infrastructure BTTP complète et fonctionnelle. La capacité d'accueil pourrait cependant être augmentée en ajoutant davantage de cœurs et de mémoire vive.

CLÉMENT M.

REALISATION DU PROJET

MISE EN PRODUCTION

CONFIGURATION DOCKER

Une fois notre VPS acquis, nous avons commencé la mise en production à l'aide de Docker. Docker nous permet de compartimenter le système à notre guise dans ce que l'on appelle des conteneurs, et d'installer une politique de sous-réseau virtuels également. Pour cette installation nous avons décidé la configuration et la mise en production de 6 conteneurs, et de 2 sous-réseaux.

- **Sous-réseau base de données** (sous-réseau comportant tous les conteneurs demandant un accès à la base de données)
 - **Conteneur application** - application bhttp serveur écoutant les connexions sur le port 1414
 - **Conteneur mysql** - contenant l'unique base de données
 - **Conteneur phpmyadmin** - pour administrer la base depuis une interface web
 - **Conteneur web** - site internet du projet
- **Sous-réseau SWAG** (sous-réseau comportant tous les conteneurs possédant un accès web)
 - **Conteneur phpmyadmin** - accessible via l'url : bd.bhttp.resrc.club
 - **Conteneur web** - accessible via l'url : bhttp.resrc.club
 - **Conteneur portainer** - nous permet d'administrer les différents conteneurs et diverses autres opérations, accessible via l'url : admin.bhttp.resrc.club
 - **Conteneur swag** - s'occupe de l'accès web, il s'occupe également de générer et de gérer notre certificat SSL garantissant la sécurité des communications via le protocole HTTPS. Il permet d'utiliser plusieurs sous-domaines pour donner accès aux interfaces web des différents conteneurs du sous-réseau SWAG, grâce au Nginx Reverse Proxy qui va s'occuper de la redirection du trafic.

Tous les conteneurs ont un accès direct à internet sauf le conteneur mysql pour des questions de sécurité, les seuls accès à la base se font par l'intermédiaire des conteneurs phpmyadmin, web ou application.

REALISATION DU PROJET

MISE EN PRODUCTION

INTÉGRATION DE LA BASE DE DONNÉES

ADMINISTRATION

Concernant les comptes pour l'accès à la base et/ou son administration. Nous avons 2 comptes :

- Un compte `'root'` accessible via l'interface web du conteneur phpmyadmin qui donne accès à toutes les bases de données présentes sur le conteneur mysql.
- Un autre compte `'bd'` accessible que via les conteneurs web et application (pour des raisons de sécurité encore une fois), ce compte n'a que les droits sur la base de données btpp et a les autorisations de modification, de récupération, de création et de suppression sur les données.

INTÉGRATION DES SCRIPTS SQL

L'intégration des scripts SQL précédemment créés (création et population) est réalisé par l'intermédiaire de phpMyAdmin qui propose une interface d'importation et d'exportation de scripts SQL.

CONNEXION AVEC LES APPLICATIONS

Les applications se connectent à la base de données via un utilisateur fait sur mesure qui se nomme `'bd'` comme expliqué précédemment. Grâce à phpMyAdmin nous avons pu restreindre la connexion de ce compte aux adresses IPs des applications en ayant besoin (ici, web et application)

REALISATION DU PROJET

FINALISATION

TESTS

Dans la phase finale du projet, nous avons essayé de réaliser un maximum de tests afin de garantir la viabilité de notre solution. Nous avons surtout concentrés ces tests sur la bibliothèque du protocole puisque beaucoup d'autres programmes de notre projet en dépendent.

CLÉMENT M.

REALISATION DU PROJET

FINALISATION

SCRIPTS DE DÉMONSTRATION

Afin de démontrer le fonctionnement de notre solution, nous avons développé trois scripts de tests à exécuter :

- [chronos.sh](#) : lance un timer qui calcule le temps écoulé depuis l'ouverture du terminal
- [telechargement.sh](#) : permet de télécharger un document en lui précisant le chemin de celui-ci
- [git-push.sh](#) : push automatique sur github

Ces scripts bash nous ont permis de tester les échanges de données avec le protocole. C'est pour cette raison que l'on a codé des scripts simples, bien que leur utilité soit limitée pour le moment. Un script bash est un fichier texte contenant des commandes Shell, exécutable par un interpréteur (ici, le programme bin/bash). Les scripts bash sont en temps normal lancés en ligne de commande, mais dans le cadre de ce projet, ils seront envoyés à partir de l'application mobile au PC distant afin qu'ils puissent être exécutés sur un terminal.

BILAN

RÉUSSITES

Ce projet est pour nous un accomplissement qui nous a apportés énormément. Nous avons tous fait de gros progrès dans nos domaines respectifs et sur des technologies que nous n'avions pas pu aborder lors de notre DUT. Nous avons pu aussi travailler sur des langages modernes comme le Dart et comprendre qu'il est important de toujours être à jour sur les nouvelles technologies. De plus, le fait de travailler sur de nombreuses technologies dans un cadre plus poussé que celui dans lequel nous évoluons habituellement nous a obligé à lire beaucoup de documentation. Nous avons aussi dû faire un grand travail sur le débogage de l'application afin de le rendre autonome. C'était très important pour nous, car le projet était conséquent et un débogage pas à pas nous aurait ralenti. Ce projet nous a donc permis à tous de continuer à développer nos compétences tant en informatique que sociale pour ce qui est du travail en groupe.

ROBIN

DIFFICULTÉS

Bien que nous avons atteint la majorité de nos objectifs, nous avons tout de même rencontré un certain nombre de difficultés. Nous nous sommes rapidement rendu compte que notre projet était ambitieux et pointu au niveau technique. Malgré les nombreux avertissements de notre professeur référent Mr. Foughali, nous avons décidé de poursuivre le projet car nous avons une projection complète du projet. En effet, nous sommes convaincus que celui-ci sera utile à long terme et pour une large communauté mêlant experts de l'informatique et débutants dans le domaine. Pour cette raison, nous allons continuer à travailler sur le projet en dehors du cadre du PJS4 afin d'y apporter des améliorations. Nous souhaitons développer de nouvelles versions du système, sur lesquelles nous aurons le plaisir de travailler durant notre temps libre.

SARA

BILAN

DIFFICULTÉS

Une autre difficulté que nous avons difficilement surmonté est la mise en commun de nos trois produits. En effet, notre projet est une solution complète qui demande la liaison de nos trois produits afin d'en assurer le bon fonctionnement. Cependant, le protocole étant le cœur de projet, il demandait plus de temps à développer que les autres produits. Nous avons donc eu des moments de creux, dans lesquels les équipes web et mobile avaient fini leur mission tandis que l'équipe protocole était encore en phase de développement. Evidemment, nous leur venons en aide lorsqu'ils en ont besoin, mais nous n'avions plus de tâches à accomplir à proprement parler. Une fois le protocole finalisé, l'intégration du protocole à l'application mobile a été difficile dans la mesure où nous avons utilisé des langages différents pour ces deux produits, ce qui posait des problèmes de compatibilité. Néanmoins, nous avons réussi à contourner les problèmes, bien que cela nous a fait perdre un peu de temps.

BILAN

AMÉLIORATIONS

RENFORCEMENT DE L'INFRASTRUCTURE

Comme nous sommes sur un VPS peu cher donc peu puissant, nous pouvons imaginer migrer notre infrastructure vers un serveur virtuel plus puissant. Bien qu'il coûtera plus cher, il nous permettra de stocker plus de données et de gérer plus de flux d'échanges. Il s'agit évidemment d'une option que l'on envisagera une fois que le projet ait été déployé en entier et qu'il soit un peu plus connu du grand public.

PROFESSIONNALISATION

TESTS UNITAIRES

Dans le but de professionnaliser notre solution, nous devons effectuer plus de tests unitaires, avec notamment plus de scripts de test. Nous pourrions par exemple proposer des scripts en python, ou dans d'autres langages de programmation pour vérifier que l'échange de messages se fait correctement.

BILAN

AMÉLIORATIONS

PROFESSIONNALISATION

PLANIFICATION DES VERSIONS

Dans la mesure où nous nous projetons sur un projet à long terme, l'organisation et la répartition des tâches sont essentielles à la bonne évolution de notre projet. En effet, dans le cadre du PJS4, nous avons dû finaliser le projet dans un temps limité, ce qui nous a contraints à passer sur certaines phases rapidement. Ayant autant de temps que nous désirons désormais, nous pouvons nous attarder sur les phases qui demandent de la réflexion et en discuter davantage entre nous. C'est pour cette raison que l'on peut, par exemple, organiser des réunions hebdomadaires afin de faire le point sur l'avancement du projet d'une part, et de discuter des futures améliorations d'autre part.

Afin de suivre le projet et son avancement, Github Desktop est un excellent outil qui nous a fortement servi dans le cadre du PJS4. Nous continuerons à l'utiliser car il est très efficace et permet d'éviter les problèmes liés à la perte de code ou au contrôle de version.

Enfin, nous nous attribuerons de nouvelles tâches une fois qu'une phase de développement a été finalisée. Pour éviter les périodes creuses, nous devons nous assurer que chaque membre de l'équipe ait une mission à accomplir. Avec le déploiement et l'évolution de notre projet, nous aurons de nouvelles responsabilités à gérer, comme de la maintenance de site par exemple. Afin de rester organisé, il serait judicieux d'avoir un tableau Trello par exemple, dans lequel chacun note son avancement, précise les problèmes rencontrés, les solutions apportées etc.

BILAN

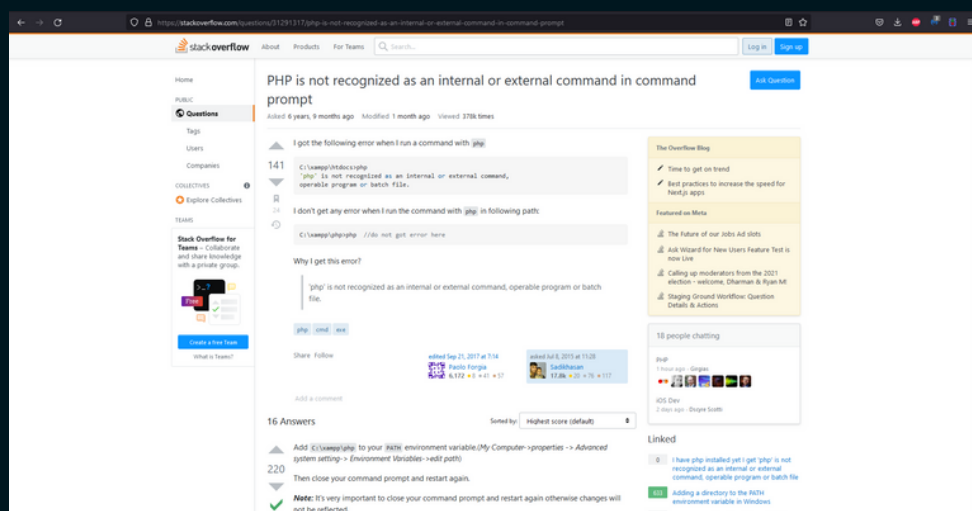
AMÉLIORATIONS

DÉVELOPPEMENT DE LA COMMUNAUTÉ

DÉVELOPPEMENT DU MARKETPLACE

Dans le futur, nous aimerions développer davantage le marketplace en ajoutant manuellement des scripts publics dans un premier temps. Ces scripts seront visibles à la fois sur l'application et sur le site web. Sachant que l'on peut consulter le code source des scripts publics sur le site web, nous invitons de potentiels contributeurs à y apporter des modifications et des améliorations. De cette manière, nous ferons connaître notre projet et ce dernier gagnera en ampleur.

Sur le site web, nous pourrions ajouter la fonctionnalité, pour les utilisateurs abonnés, d'ajouter des scripts publics, à l'image de Github. Nous voulons faire du site web et de l'application un lieu de partage et d'échanges entre les développeurs et les non initiés. Un espace d'entraide et de solidarité, qui permettra d'affiner la ligne qui sépare les informaticiens des débutants dans le domaine. Nous pourrions même ajouter un onglet dans le site web qui permettra d'échanger sur le sujet, à la manière de Stack Overflow, une plateforme de questions réponses largement répandue chez les développeurs.



Capture d'écran du site StackOverflow

SARA

BILAN

AMÉLIORATIONS

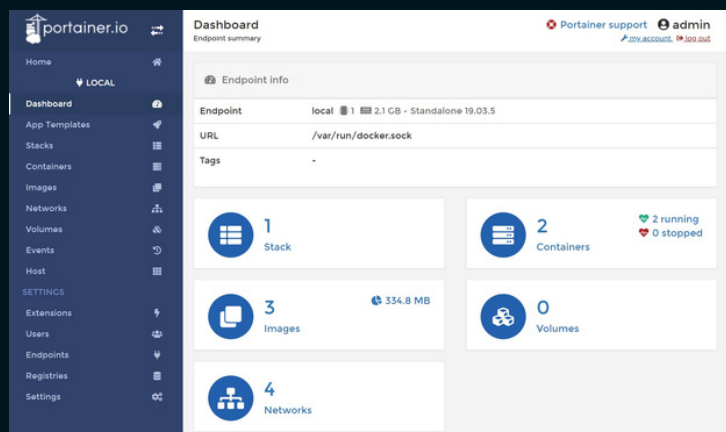
DÉVELOPPEMENT DE LA COMMUNAUTÉ

DÉVELOPPEMENT DU MARKETPLACE

Nous souhaitons développer une communauté qui tirera sa force de la diversité des utilisateurs. En effet, nous sommes convaincus que le futur du projet réside dans le partage des connaissances des utilisateurs, peu importe leur niveau. C'est pour cette raison qu'il nous semble essentiel que cette communauté ait un espace dédié sur notre site web.

Améliorations de l'accessibilité

Comme nous l'avons dit précédemment, nous avons développé une application en ligne de commande qui permettra aux ordinateurs de jouer le rôle de client. Pour l'instant, celle-ci n'est pas intuitive pour les débutants et reste plutôt réservée aux développeurs, qui sont à l'aise avec l'utilisation des terminaux. Dans le futur, nous voudrions développer l'aspect graphique du CLI, à l'image de Portainer, l'interface graphique de Docker. Voici des exemples ci-après :



Capture d'écran du GUI Portainer

SARA

BILAN

AMÉLIORATIONS

DÉVELOPPEMENT DE LA COMMUNAUTÉ

AMÉLIORATION DE L'ACCESSIBILITÉ

En effet, un utilisateur qui veut utiliser notre système via un ordinateur plutôt qu'en passant par l'application mobile devra utiliser notre CLI (Command Line Interface). Or, dans notre philosophie de rendre notre projet accessible à tous, il n'est pas facile pour un utilisateur non initié d'utiliser notre système ainsi. C'est pour cette raison que nous voulons le rendre accessible en intégrant un GUI (Graphic User Interface) en plus de notre CLI. A l'image de l'application mobile, ce GUI permettra de faire jouer le rôle de client à un PC et envoyer des demandes d'exécution de scripts à un autre PC distant.

CONCLUSION

Pour conclure, c'est un projet dont nous sommes très fiers. Nous sommes heureux d'avoir pu effectuer un projet qui nous plaisait réellement et d'une grande envergure, qui pourrait ressembler à ce qu'on rencontrera dans le monde du travail.

Ce projet nous a permis de nous améliorer sur plusieurs points. Dans un premier temps, nous avons tous progressé sur le plan technique, chacun dans son domaine ainsi que sur les parties des autres lorsqu'une aide était requise. Cela nous a permis d'apprendre d'une manière plus poussée certaines technologies et que nous avons peu ou pas explorés lors de nos cours.

Ensuite, nous avons aussi progressé sur la gestion de projet. Pour réaliser un projet complexe, divisé en 3 parties, il nous a fallu une très bonne organisation. Cette dernière a été primordiale. Nous avons donc appris l'importance de séparer les tâches et les missions en fonction des forces de chacun, mais aussi qu'une mise en commun régulière était fondamentale pour avancer au même rythme et être le plus efficace possible.

Enfin, ce projet est pour nous plus qu'un simple PJS4, nous souhaitons tous continuer à travailler dessus afin de pouvoir le partager plus tard. Nous pensons que ce logiciel peut avoir un bel avenir et aider beaucoup de monde. Cette motivation nous vient du fait que notre projet réponde à un problème que nous avons, il nous sera fort utile dans le futur, tout comme d'autres personnes rencontrant le même problème. De plus, ce projet personnel est pour nous un bon moyen de nous démarquer professionnellement avec un travail abouti à présenter lors de potentielles recherches d'écoles ou d'emplois.

REMERCIEMENTS

Tout d'abord, un grand merci à toute l'équipe d'avoir fourni autant d'efforts dans le projet et d'y avoir cru jusqu'au bout. Merci pour le bon déroulement du projet, de la soutenance, et tout particulièrement pour l'ambiance du groupe qui a positivement affecté notre travail.

Nous tenons à remercier notre professeur référent, Monsieur Foughali, de nous avoir suivi lors de la phase de développement et de nous avoir mis en garde des risques du projet. Bien que vous ayez eu des doutes, vos remarques nous ont permis de nous mettre en garde de certains risques que l'on a pu contourner avec succès. Vous nous avez été d'une grande aide.

Merci à Madame Mayer, présente lors de la soutenance, de nous avoir écouté et de vous être intéressée à notre projet malgré sa complexité.

Un grand merci à Monsieur Brette, qui a inspiré le nom du projet ainsi que son sujet.

Nous adressons nos sincères remerciements à l'ensemble de l'équipe pédagogique de l'IUT de Paris, vous avez participé au bon déroulement de nos deux années de DUT et tout cela n'aurait pas été possible sans vous.

TOUTE L'EQUIPE BTTP

ANNEXE

Vous trouverez ci joint le lien github vers notre projet. Dans le dossier projet, vous trouverez un manifeste qui détaille les termes techniques de notre rapport. Nous espérons sincèrement qu'il vous a plu :

<https://github.com/tensaiji/PJS4>