

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique
Centre Universitaire Nour Bachir El Bayadh



Institut des sciences
Département de Technologie
PROJET DE FIN DE CYCLE
POUR L'OBTENTION DU DIPLOME DE MASTER
ELECTRONIQUE DES SYSTEMES EMBARQUES

INTITULE :

Pilotage à distance d'un système de contrôle pour un laboratoire
distant

Soutenu le: 27/09/2020

Présenté par:

Mr. HEMIDI Benameur

Mr. BENALI Ghiet

Devant le Jury composé de:

Dr. NOUR Mohammed	CU NOUR Bachir EL Bayadh	Président
Dr. BENDELHOUM Mohammed S.	CU NOUR Bachir EL Bayadh	Examineur
Dr. ROUISSET Mehdi	CU NOUR Bachir EL Bayadh	Examineur
Dr. BENALI Abdelkrim	CU NOUR Bachir EL Bayadh	Encadrant
Dr. BELKHEIR Mohammed	CU NOUR Bachir EL Bayadh	Co-Encadrant1
Dr. FARAH Said	Chercheur, Laboratoire XLIM, Limoges, France	Co-Encadrant2
Dr. DJELDJLI Youcef	CU NOUR Bachir EL Bayadh	Invité

Année Universitaire 2019 / 2020

Remerciements

***Nous tenons** à remercier en premier lieu mon dieu Allah tout puissant qui nous a aidé et qui nous a donné la volonté et le courage pour la réalisation de ce travail.*

***Nous tenons** à remercier tout d'abord mes encadrants, le docteur BENALI Abdelkrim, Dr. BELKHEIR Mohammed et Dr. FARAH Said de l'université de Limoges pour la patience et l'encouragement qu'ils nous ont accordé le long de ce travail, pour la confiance, la compréhension et leurs conseils pour la rédaction de ce mémoire.*

***Nous** adressons mes remerciements au président du jury Dr NOUR Mohammed et les Docteurs: ROUISSAT Mehdi, BENDELHOUM Mohammed Sofiane et Mr. DJELDJLI Youcef pour avoir acceptés d'être membres du jury en portant un intérêt particulier à notre travail.*

***Nous voulons** aussi exprimer nos vives reconnaissances envers tous les enseignants de l'institut des sciences au Centre Universitaire Nour Bachir El-Bayadh et spécialement les enseignants de la spécialité master électronique des systèmes embarqués, sans oublier nous chers collègues de promotion pour leurs aides et leurs vives encouragements pendant la réalisation du présent travail.*

Table des matières	ii
Liste des tableaux	v
Liste des figures	v
Liste des abréviations	viii
Introduction Générale	1

Chapitre I

Systèmes embarqués pilotables à distance SEPD

I.1	Introduction	3
I.2	Systèmes embarqués	3
I.2.1	Conception de systèmes embarqués	3
I.2.2	Une introduction à l'architecture des systèmes embarqués	4
I.2.3	Pourquoi l'architecture d'un système embarqué est-elle importante?	4
I.2.4	Le modèle des systèmes embarqués	6
I.2.5	Systèmes embarqués pilotables à distance SEPD	6
I.2.6	Caractéristiques principales d'un système embarqué:	6
I.2.7	Domaines d'application d'un système embarqué	7
I.3	Les travaux pratiques (TP)	7
I.3.1	Définition des travaux pratiques	7
I.3.2	Les objectifs pédagogiques d'un TP	8
I.3.3	Les interactions d'un TP	8
I.3.4	Cycle de déroulement d'un TP	9
I.3.5	Les limites d'un TP	9
I.3.6	Les travaux pratiques (TPs) et l'informatique	9
I.3.7	Laboratoires locaux assistés ou non par l'ordinateur	9
I.4	Laboratoire à distance	11
I.4.1	Laboratoire virtuel	13
I.4.2	Laboratoire hybride	13
I.5	L'apprentissage dans laboratoire d'ingénierie	13
I.6	Impact socio-économique	15
I.7	Conclusion	15

Chapitre II

Laboratoires distants : projets, caractéristiques matérielles et logicielles

II.1	Introduction	16
II.2	Projet iLAB.	16
II.2.1	Partie logicielle iLAB	17
II.2.2	Partie matérielle iLAB	18
II.3	Projet VISIR	19
II.3.1	Partie logicielle VISIR	20
II.3.2	Partie matérielle VISIR	21
II.4	Projet ISILAB	23
II.4.1	Partie Logicielle ISILAB	24
II.4.2	Partie matérielle ISILAB	25
II.5	Projet NETLAB	26
II.5.1	Partie logicielle NETLAB	27
II.5.2	Partie matérielle NETLAB	28
II.6	Projet RwmLAB	30
II.6.1	Partie logicielle RwmLAB	31
II.6.2	Partie matérielle RwmLAB	31
II.7	Laboratoire distant LABTEC	32
II.7.1	Contraintes et caractéristiques du laboratoire LEBTEC	33
II.8	Conclusion	34

Chapitre III

Projet LABTEC, conception, réalisation, discussions des résultats

III.1	Introduction	35
III.2	Cahier des charges (spécifications fonctionnelles)	35
III.3	Conception et modélisation par UML	36
III.3.1	Conception UML	36
III.3.1.2	Diagramme de classe	38
III.3.1.3	Diagramme de séquence	39
III.4	Réalisation du projet LABTEC	43

III.4.1	Architecture cible de LABTEC	43
III.4.2	Les outils d'implémentations	44
III.4.2.1	Les outils de la Plate-forme Matérielle	44
III.4.2.2	Les outils de la plate-forme logicielle	48
III.4.3	Réalisations et discussions	52
III.4.3.1	Première solution proposée, utiliser Django et node.js	52
III.4.3.2	Deuxième solution « Utilisation des bibliothèques Python »	55
III.4.3.3	Troisième solution : utilisation d'un OS (Operating System)	57
III.4.3.4	Python et le système d'exploitation (Operating System)	59
III.4.3.5	Améliorations ajoutées au système	61
III.4.3.6	Création d'un serveur web pour l'instrument de mesure	64
III.4.3.7	Authentification et autorisation	65
III.4.3.8	Modèles et bases de données	67
III.4.3.9	Architecture réseau du TP réalisé	69
III.5	Résultats, discussions et perspectives	70
III.5.1	Présentation de l'application	70
III.5.2	Discussions et perspectives	78
III.6	Conclusion	80
	Conclusion générale	81
	Bibliographie	83
	Annexe A	86
	Annexe B	88
	Annexe C	90

Liste des tableaux

Tableau III.1	Comparatif des spécifications des modèles du Raspberry Pi.	45
---------------	--	----

Liste des figures

Figure I.1 :	Modèle de systèmes embarqués	6
Figure I.2 :	interactions lors d'une séance de TP classique	8
Figure I.3 :	Laboratoire local	10
Figure I.4 :	Laboratoire local assisté par ordinateur	10
Figure I.5 :	Principe d'un laboratoire proximal	11
Figure I.6 :	Principe d'un laboratoire distant	11
Figure I.7 :	exemple laboratoire distant	12
Figure II.I:	Architecture d'iLab	18
Figure II.2 :	Architecture d'iLab basée sur plateforme ELVIS	19
Figure II.3 :	L'étudiant sélectionner l'interface instrument avec laquelle il est familier	20
Figure II.4 :	Architecture de projet VISIR	21
Figure II.5 :	Exemple SPRL complexe: VISIR, prenant en charge plusieurs étudiants simultanés	22
Figure II.6:	Projet VISIR: carte électronique utilisateur (à gauche) et matrice de commutation (droite)	23
Figure II.7 :	Interfaces utilisateur d'ISILab	23
Figure II.8 :	Carte mère (ISiBoard) avec les cartes d'expériences	26
Figure II.9 :	Architecture de NETLAB	27
Figure II.10 :	Les interfaces des instruments utilisés dans NETLAB	28
Figure II.11:	La plate-forme physique de NetLab	29
Figure II.12:	Interface de générateur de circuit de NetLab	30
Figure II.13:	Interface utilisateur de RwmLab	31
Figure II.14:	La plate-forme physique de RwmLAB	32
Figure II.15 :	La plate-forme physique de LEBTEC	33
Figure III.1:	Diagramme de cas d'utilisation	37

Figure III.2	Diagramme de classe de l'application	39
Figure III.3 :	Diagramme de séquence (inscription utilisateur)	40
Figure III.4 :	Diagramme de séquence (réalisation TP étudiant)	41
Figure III.5 :	Diagramme de séquence (configuration tp enseignant)	42
Figure III.6 :	Architecture cible de LABTEC	43
Figure III.7 :	carte Raspberry Pi 4 Modèle B	45
Figure III.8 :	carte TP et Matrice de commutation	46
Figure III.9 :	Oscilloscope PeakTech 1240	47
Figure III.10	Interfaces PeakTech 1240	47
Figure III.11	Django MVT: Architecture d'un projet	50
Figure III.12	Flux de données du système proposé	53
Figure III.13	Structure du fichier Pi_setting.js	54
Figure III.14	Architecture logicielle de la solution (première variante)	54
Figure III.15	lien entre l'HTML et Views.Py	56
Figure III.16	Le graph CFG de la fonction setting_GPIO	57
Figure III.17	Commandes d'entrée et de sortie standard (gpio)	58
Figure III.18	exemple d'utilisation du module « subprocess »	59
Figure III.19	le module subprocess dans notre système	60
Figure III.20	Développement de la fonction « carte TP »	60
Figure III.21	Changement de l'état du GPIO avec confirmation	61
Figure III.22	Commande GPIO sur la même page	62
Figure III.23	Relation entre Ajax et le serveur	63
Figure III.24	serveur Flask pour accès à l'oscilloscope PeakTEch	65
Figure III.25	Modèles et bases de données	67
Figure III.26	Structure de la base de données	68
Figure III.27	Architecture réseau du TP réalisé	69
Figure III.28	page d'accueil de l'application	70
Figure III.29	Inscription de l'utilisateur, choix du type de compte	70
Figure III.30	Interface inscription des étudiants	71

Figure III.31	Interface inscription de l'enseignant	72
Figure III.32	Page d'accueil de l'enseignant	72
Figure III.33	Interface ajouter un nouveau travail pratique TP	73
Figure III.34	Interface de dépôt du protocole du TP (enseignant)	73
Figure III.35	Interface résultats du TPs	74
Figure III.36	Interface évaluation des étudiants	74
Figure III.37	page d'accueil de l'étudiant	75
Figure III.38	Annonces de TP (étudiants)	75
Figure III.39	Interface réalisation de TP (étudiants)	76
Figure III.40	Interface de l'administrateur (page d'accueil)	77
Figure III.41	Interface de l'administrateur (Liste des utilisateurs)	77
Figure III.42	Interface de l'administrateur (Autorisations)	78

Liste des Abréviations

- **SEPD** :Systèmes embarqués pilotables à distance
- **TP** : Travaux pratiques.
- **PCI**:*Peripheral Component Interconnect.*
- **DVD- ROM** :*Digital Versatile Disk - Read Only Memory.*
- **IOT** :*Internet of Things.*
- **IOE** :*Internet of Everything.*
- **SCPI** :*Standard Commands for Programmable Instruments.*
- **USB** :*Universal Serial Bus.*
- **RS-232** :*Recommended Standard 232.*
- **RAM** :*Random-Access Memory.*
- **SDRAM** :*Synchronous Dynamic Random-Access Memory.*
- **SRAM** :*Static random access memory*
- **IEEE** :*Institute of Electrical and Electronics Engineers.*
- **CNA** :*Convertisseur Numérique Analogique.*
- **HTML** :*HyperTextMarkupLanguage.*
- **CSS**:*Cascading Style Sheet.*
- **AJAX** :*Asynchronous JavaScript and XML.*
- **PHP** : *HypertextPreprocessor.*
- **WSGI** :*Web Server Gateway Interface.*
- **JSON** :*JavaScript Object Notation.*
- **XML** :*ExtensibleMarkupLanguage.*
- **LMS** : Learning Management System
- **SGBD** : système de gestion de base de données
- **WEB** : World Wide Web
- **LABTEC** : *LABORATOIRE DE TECHNOLOGIE EL BAYADH*
- **UI/UX** :*User Interface/ User Experience*
- **LAN** : *Local Area Network*

- *LXI :LANeXtensions for Instrumentation*
- *PXI : PCI eXtensions for Instrumentation*
- *GUI : Graphical user interface*
- *GPIB :General Purpose Interface Bus*
- *SVG : Scalable Vector Graphics*
- *ORM :Object-relational mapping*
- *SQL :Structured Query Language*

Introduction Générale

La pandémie de Corona a envahi le monde et elle a eu le pouvoir de le contraindre à une série de changements immédiats et implicites couvrant différents niveaux, économie, politique, médecine, ainsi que les domaines de l'éducation, de sorte que cette pandémie a imposé - dans le monde arabe du moins - de nouvelles méthodes éducatives, qui ont également produit de nouvelles façons de recevoir des informations. Où l'e-learning était un phénomène académique résultant de l'existence de cette pandémie

L'e-learning a trouvé divers soutiens, car de nombreux appels en faveur de la transformation numérique dans le processus éducatif ont été soutenus, afin de suivre le rythme du développement temporel et technologique, ainsi le support technologique avec toutes ses plates-formes est devenu un dépositaire de beaucoup de connaissances et de sciences, donc tout le monde doit en profiter dans l'industrie.

Grâce aux avancés technologiques en informatique, en réseau et en communication, de nouveaux moyens d'expérimentation ont été instaurés à savoir : l'expérimentation virtuelle et l'expérimentation à distance. Dans l'expérimentation virtuelle, l'environnement des expériences est un modèle mathématique (simulation) sur un ordinateur. Dans l'expérimentation à distance, l'expérimentateur se situe hors du laboratoire (à distance) et manipule les équipements du laboratoire réel à travers un réseau de communication généralement Internet.

Ce nouveau concept qui permet d'offrir de réels services, reste encore limité à cause des problèmes organisationnels, humains, et particulièrement techniques. Il est aisé de présenter un cours en ligne, par contre, il est beaucoup plus difficile de donner aux apprenants les moyens de manipuler des dispositifs technologiques (instruments de laboratoire) à distance dans un cadre pédagogique.

Notre travail consiste à concevoir et mettre en œuvre un environnement d'apprentissage à distance, qui assure la manipulation d'une plate-forme matérielle (vrais dispositifs techniques) à travers l'accès à une plateforme LMS, dans le domaine de l'enseignement de l'Electronique. Cet environnement doit remplir les mêmes objectifs des laboratoires classiques, minimiser les contraintes pédagogiques en réduisant le

nombre et le coût du matériel utilisé dans les travaux pratiques, adapter l'enseignant aux difficultés liées au contrôle matériel ainsi qu'à la communication avec les étudiants pour mieux évaluer les TPs.

Ce mémoire est décomposé en trois chapitres, qui sont brièvement introduits dans le paragraphe suivant :

- **Chapitre 1** : Nous allons introduire quelques concepts des systèmes pilotables à distance (SEPD) ainsi que les notions générales sur les systèmes embarqués.
- **Chapitre 2** : Dans ce chapitre, nous allons décrire les différents concepts de notre projet du labo distant que nous avons baptisé : LABORATOIRE DE TECHNOLOGIE EL BAYADH (LABTEC), ainsi que les autres solutions existantes.
- **Chapitre 3** : Ce chapitre détaille la conception de notre système (matérielle et logiciel), la modélisation UML de notre système, la réalisation du système en exposant les résultats obtenus et les problèmes rencontrés ainsi que quelques perspectives d'amélioration de notre prototype.

Chapitre I

Systèmes embarqués pilotables à distance SEPD

I.1 Introduction

Les systèmes embarqués sont des systèmes électroniques et informatiques autonomes, soumis à des fortes contraintes énergétiques et temps réel.

Ce chapitre présente au lecteur une vue générale sur le monde des systèmes embarqués qui ont envahi notre vie quotidienne. Nous expliquerons les principes de base d'un système embarqué temps réel, en passant en revue son évolution dès son apparition jusqu'aux technologies de nouvelle génération comme l'Internet des Objets (IOT), les systèmes cyber-physiques et le Big Data.

I.2 Systèmes embarqués

Un système embarqué peut être défini comme un système électronique et informatique autonome, qui est dédié à une tâche bien précise, possédant des ressources d'ordre spatial (taille limitée) et énergétique (consommation restreinte) limitées. Le terme de « Système Embarqué » désigne aussi bien le matériel que le logiciel utilisé. Les systèmes embarqués exécutent des tâches prédéfinies et ont un cahier des charges contraignant à remplir,

I.2.1 Conception de systèmes embarqués

Lors de l'approche de la conception d'architecture de systèmes embarqués à partir d'un point d'ingénierie des systèmes, plusieurs modèles peuvent être appliqués pour décrire le cycle de conception d'un système embarqué. La plupart de ces modèles sont basés sur une ou une combinaison des modèles de développement :

- Le modèle du big-bang, dans lequel il n'y a pratiquement pas de planification ou de processus en place avant et pendant le développement d'un système.

- Le modèle de code et de correctif, dans lequel les exigences du produit sont définies mais pas formelles des processus sont en place avant le début du développement.

- Le modèle en cascade, dans lequel il existe un processus de développement d'un système par étapes, où les résultats d'une étape passent à l'étape suivante

- Le modèle en spirale, dans lequel il existe un processus pour développer un système par étapes, et tout au long des différentes étapes, des retours sont obtenus et réintégrés dans le processus.

I.2.2 Introduction à l'architecture des systèmes embarqués

L'architecture d'un système embarqué est une abstraction du dispositif embarqué, ce qui signifie qu'il s'agit d'une généralisation du système qui ne montre généralement pas la mise en œuvre détaillée des informations telles que le code source du logiciel ou la conception du circuit matériel. Au niveau architectural, les composants matériels et logiciels d'un système embarqué sont plutôt représentés comme une composition d'éléments en interaction. Les éléments sont des représentations de matériel et / ou logiciel dont les détails d'implémentation ont été résumés, ne laissant que le comportement et les informations sur les relations. Les éléments architecturaux peuvent être intégrés en interne dans le périphérique intégré, ou existent à l'extérieur du système intégré et interagissent avec les éléments. En bref, une architecture embarquée comprend des éléments du système embarqué, des éléments interagissant avec un système embarqué, les propriétés de chacun des éléments individuels, et les relations interactives entre les éléments. Les informations au niveau de l'architecture sont représentées physiquement sous la forme de structures.

Une structure est une représentation possible de l'architecture, contenant son propre ensemble d'éléments représentés, de propriétés et d'informations sur les relations. Une structure est donc une «instance» du matériel et des logiciels du système au moment de la conception et / ou de l'exécution, compte tenu d'un environnement particulier et d'un ensemble d'éléments donné. Puisqu'il est très difficile pour un «instance» de capturer tous les complexités d'un système, une architecture est généralement composée de plus d'une structure. Toutes les structures d'une architecture sont intrinsèquement liées les unes aux autres,

I.2.3 Pourquoi l'architecture d'un système embarqué est-elle importante?

En bref, une architecture de systèmes embarqués peut être utilisée pour résoudre ses défis dès le début du projet. Sans définir ni connaître aucun détail de mise en œuvre interne, l'architecture d'un appareil embarqué peut être le premier outil à analyser et à utiliser en tant que plan de haut niveau définissant l'infrastructure d'une conception, les options de conception possibles et les contraintes de conception. Ce qui rend l'approche architecturale si puissante, c'est sa capacité à communiquer rapidement un design à une variété de personnes avec ou sans formation technique, agissant même comme une fondation dans la planification du projet ou la conception d'un appareil. Parce qu'elle définit clairement les exigences du système, une architecture peut servir de base solide

pour analyser et tester la qualité d'un appareil et ses performances dans diverses circonstances. De plus, si elle est comprise, créée et exploitée correctement, une architecture peut être utilisée pour estimer avec précision et réduire les coûts grâce à sa démonstration des risques liés à la mise en œuvre des différents éléments, permettant d'atténuer ces risques. Enfin, les différentes structures d'une architecture peuvent ensuite être utilisées pour concevoir de futurs produits avec des caractéristiques, permettant ainsi la réutilisation des connaissances de conception et entraînant un futur coûts de conception et de développement.

Comprendre l'architecture d'un système embarqué est une composante essentielle de bonne conception du système. En effet, en plus des avantages énumérés ci-dessus:

1. Chaque système embarqué possède une architecture, qu'elle soit ou non documentée, parce que chaque système embarqué est composé d'éléments en interaction (qu'ils soient matériels ou logiciels). Avant de commencer le développement, il est impératif de prendre le contrôle de la conception en définissant d'abord l'architecture du système.

2. Parce qu'une architecture intégrée capture diverses vues, qui sont des représentations du système, elle est un outil utile pour comprendre tous les principaux éléments du système et leurs rôles et fonctionnalités attendus, .. Chaque élément du système interagit avec un autre élément d'une manière ou d'une autre. Sans comprendre le rôle, la fonctionnalité, les performances etc. d'un élément, il serait difficile de déterminer comment le système fonctionnerait. . Même si les structures architecturales sont brutes et informelles, c'est encore mieux que rien. Comme l'architecture transmet d'une manière ou d'une autre les composants critiques d'une conception et leurs relations entre eux, elle peut fournir aux membres du projet des informations sur les exigences du systèmes embarqué et comment un tel système peut être construit avec succès

I.2.4 Modèle des systèmes embarqués

Tous les systèmes embarqués partagent une similitude au plus haut niveau; c'est-à-dire qu'ils ont tous au moins une couche (matériel) et/ou toutes les couches logiciel système et logiciel d'application) dans lequel tous les composants y intègrent. La couche matérielle contient tous les principaux composants physiques situés sur une carte embarquée, tandis que les couches système et logiciel d'application contiennent tous les logiciels qui font fonctionner le système embarqué. Ce modèle de référence est essentiellement une

représentation en couches (modulaire) d'une architecture de systèmes dont une structure architecturale modulaire peut être dérivée [1].

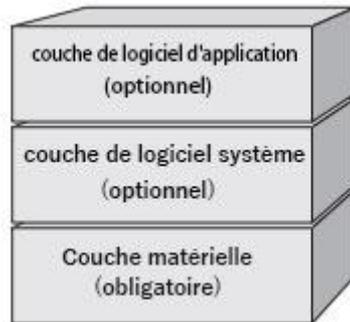


Figure I.1: Modèle de systèmes embarqués [36]

I.2.5 Systèmes embarqués pilotables à distance SEPD

Les termes de « système embarqué pilotables à distance » désignent aussi bien le matériel que le logiciel utilisés. Pour concevoir un système embarqué qui peut être contrôlé à distance, il faut généralement combiner des compétences en électronique, en informatique industrielle et en automatique. Les Systèmes Embarqués pilotables à distance sont présents dans la très grande majorité des équipements, et des systèmes actuels

I.2.6 Caractéristiques principales d'un système embarqué

Les principales caractéristiques d'un système embarqué sont les suivantes :

- C'est un système principalement numérique.
- Il met en œuvre généralement un processeur.
- Dédié pour une application spécifique.
- Coût réduit.
- Espace restreint (volume, capacité mémoire).
- Capacité de calcul appropriée et adaptée.
- Exécution temps réel.
- Fiabilité et sécurité de fonctionnement.
- Consommation d'énergie maîtrisée, voire très faible en cas d'utilisation sur batterie.

I.2.7 Domaines d'application d'un système embarqué

Les grands secteurs de l'embarqué concernent les domaines suivants :

- systèmes de contrôle: automobile, process chimique, process nucléaire, système de navigation...

- Traitement du signal : radar, sonar, compression vidéo.

- Communication et réseaux : transmission d'information et commutation, téléphonie, Internet, serveur de temps, pare-feu.

- Électroménager : télévision, four à micro-ondes, machine à laver ...etc.

- Informatique : disque dur, lecteur de disquette, etc.

- Équipement médical

I.3 Travaux pratiques (TP)

Toute formation, qu'elle soit initiale (du primaire au supérieur) ou continue, a besoin de proposer des sessions de formation où l'étudiant est confronté au monde réel afin qu'il mette en pratique ses connaissances et son savoir-faire et qu'on puisse juger de son opérationnalité [2]. Les cours représentent souvent la première étape et sont chargés d'aborder des théories et des concepts de base. Ensuite viennent les Travaux Dirigés (TD), à mi-chemin entre la théorie et la pratique, pour approfondir ces concepts en favorisant un travail avec exemples et exercices courts. Enfin, les Travaux Pratiques (TP) offrent aux étudiants une confrontation entre ce qui a été abordé en cours, travaux dirigés et la réalité.

I.3.1 Définition des travaux pratiques

-Les travaux pratiques sont définis par « l'activité qui consiste, pour les étudiants, à effectuer une ou plusieurs expérimentations, c'est-à-dire des manipulations de dispositifs expérimentaux avec l'objectif de mettre en évidence un phénomène » [3].

-Les travaux pratiques sont une forme d'activité d'apprentissage qui se fait dans un laboratoire et qui permet à un groupe d'étudiant d'expérimenter les principes théorique apprise en suivant les consignes de l'enseignant.

I.3.2 Les objectifs pédagogiques d'un TP

Les objectifs pédagogiques des travaux pratiques sont [4] :

- Fournir des illustrations et des démonstrations des principes enseignés et donc une meilleure assimilation des apprenants.

- Motiver les étudiants et focaliser les interactions entre apprenants et entre apprenants enseignants.
- Développer des compétences pratiques considérées comme importantes d'un point de vue professionnel.
- Développer des compétences de travail collaboratif en équipe.
- Introduire les étudiants dans la communauté pratique des scientifiques

I.3.3 Les interactions dans un TP

Dans un TP, les apprenants peuvent manipuler et visualiser le matériel (le dispositif). La séance de TP est le lieu d'un échange privilégié avec l'enseignant. Les apprenants posent plus facilement des questions profitant du fait d'être encadrés en petits groupes. Pour l'enseignant, le TP lui permet d'évaluer les réactions des apprenants, leur capacité de travail et leur évolution soit d'une façon individuelle ou travailler en groupe. La Fig.1.2 illustre les interactions lors d'une séance de TP :

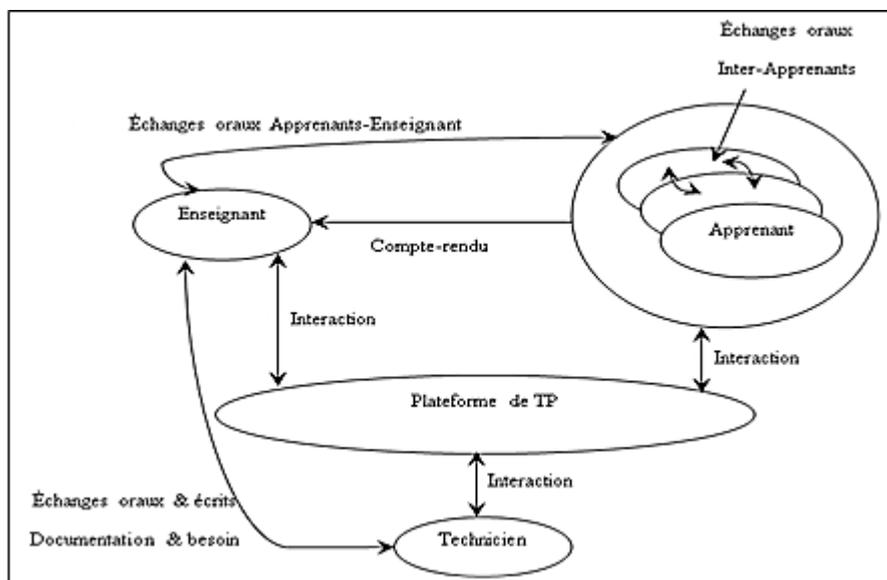


Figure I.2: Interactions lors d'une séance de TP classique.[5]

I.3.4 Cycle de déroulement d'un TP

Un TP classique est divisé en trois étapes :

- Une phase de préparation : elle consiste en une lecture approfondie du texte exposant les notions théoriques utiles et la manipulation.
- La séance de TP : elle débute généralement par un rappel des objectifs du TP et une présentation du matériel puis la manipulation réalisée par l'apprenant.

- La phase de rédaction d'un compte-rendu.

I.3.5 Les limites d'un TP

Les travaux pratiques classiques présentent quelques limitations intrinsèques :

- Le temps: les tranches horaires limitées dans l'emploi du temps de TP peuvent ne pas avoir lieu dans l'ordre le plus approprié pour certains groupes. En outre, les contraintes des emplois du temps font que certains TPs arrivent avant le cours.

- La documentation : dans le pire des cas, le sujet est donné au moment de la séance et est parfois indisponible en dehors des séances de TP.

- L'équipement : certains TPs nécessitent un équipement lourd et onéreux que l'on ne peut pas dupliquer.

I.3.6 Les travaux pratiques (TPs) et l'informatique

L'utilisation de l'informatique dans le cadre des travaux pratiques s'est banalisée. La puissance de calcul et la capacité élevée de stockage des données, l'une des caractéristiques les plus importantes de l'ordinateur est la diversité de ses usages. Lors d'un TP, il peut être utilisé comme outil de calcul, de traitement de données, de support multimédia, ou même comme assistant à l'expérimentation (avec la possibilité de récupération automatique de données à partir des instruments). Par analogie avec la CAO (conception), la DAO (Dessin), la FAO (Fabrication) et l'EAO (Enseignement), l'utilisation de l'ordinateur en tant qu'outil d'aide et d'assistance (AO : Assisté par Ordinateur) a donné naissance au terme ExAO (Expérimentation Assistée par Ordinateur).[44]

I.3.7 Laboratoires locaux assistés ou non par l'ordinateur

En générale, conduire une expérimentation c'est la réaliser sur place dans un lieu qu'on appelle laboratoire. Lorsque l'ordinateur était encore absent des salles de travaux pratiques, la manipulation était directe (Figure I.3). On appelle « laboratoire local », cette situation où aucun média ne joue l'interface entre l'homme et l'expérimentation. Cette façon de procéder existe toujours, car toutes les expérimentations ne nécessitent pas forcément l'emploi d'un ordinateur, mais elle tend à disparaître.[44]

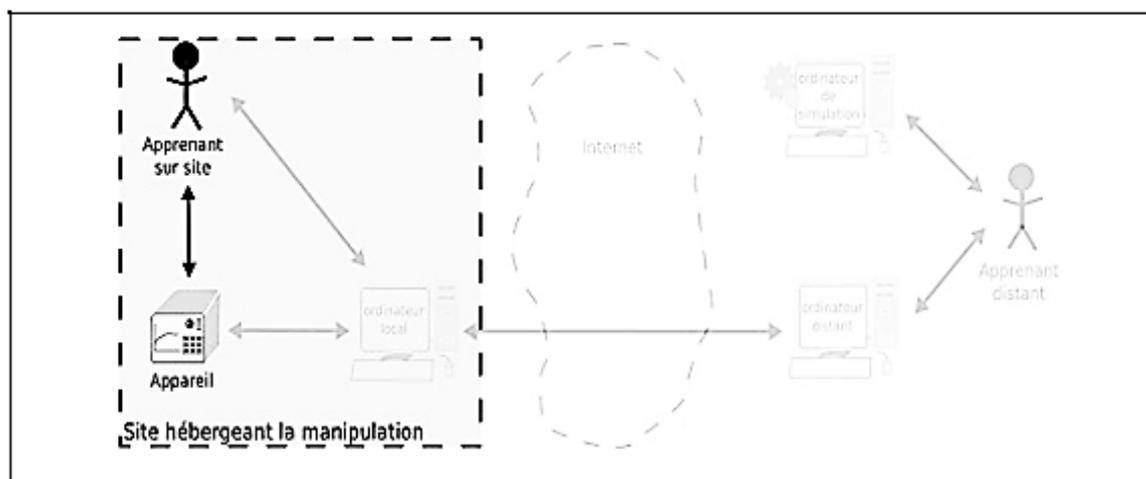


Figure I.3: Laboratoire local [5].

En effet, l'ordinateur apporte bien souvent une nouvelle dimension à la manipulation considérée (visualisation, capacité d'utilisation, adaptation). On parle alors de « laboratoire local assisté par ordinateur » (Figure I.4).

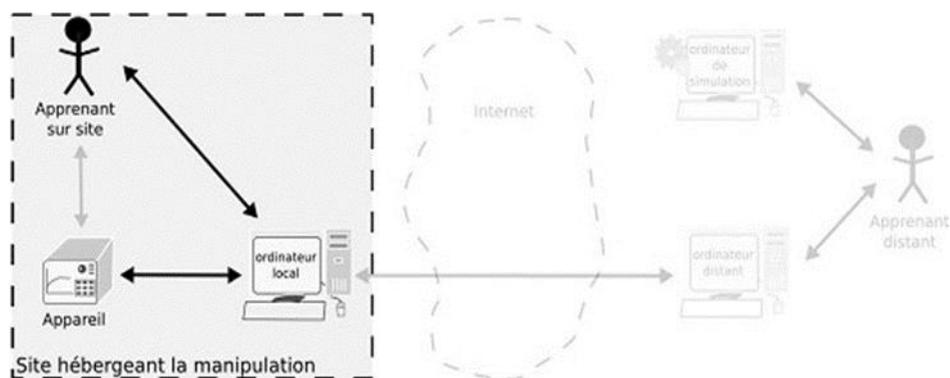


Figure I.4: Laboratoire local assisté par ordinateur [5].

La manipulation est direct dans le sens où il est encore possible d'agir physiquement avec les dispositifs, car l'ordinateur ne sait pas ou ne peut pas gérer entièrement la manipulation. Une panne peut également surgir mais la présence au même endroit et en même temps des dispositifs et des apprenants permet de surmonter ce genre de problème.

I.4 Laboratoires à distance

Les laboratoires distants permettent aux utilisateurs d'effectuer des expériences et des tâches de laboratoire sur Internet sans être à proximité de l'équipement réel. Dans un laboratoire proximal traditionnel, l'utilisateur interagit directement avec l'équipement en effectuant des actions physiques (par exemple en manipulant avec les mains, en appuyant sur les boutons, en tournant les boutons) et en recevant un retour sensoriel (visuel, audio et tactile).

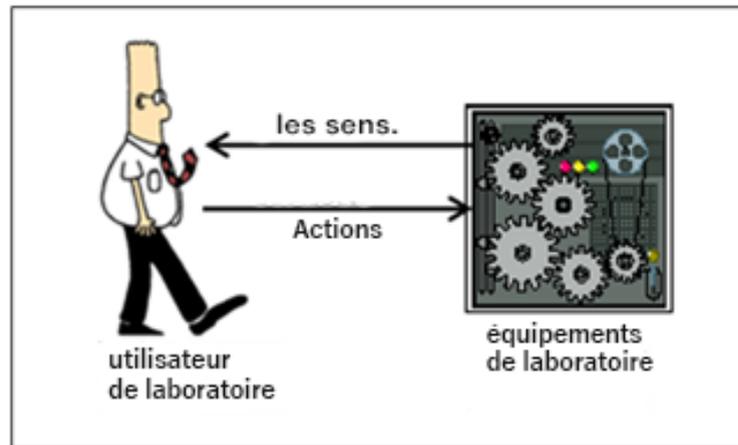


Figure I.5 : Principe d'un laboratoire proximal

Dans un laboratoire distant, cette même interaction a lieu à distance avec l'aide de l'infrastructure distante. Il s'agit d'une nouvelle couche qui se situe entre l'utilisateur et l'équipement de laboratoire. Il est chargé de transmettre les actions de l'utilisateur et de recevoir les informations sensorielles de l'équipement.

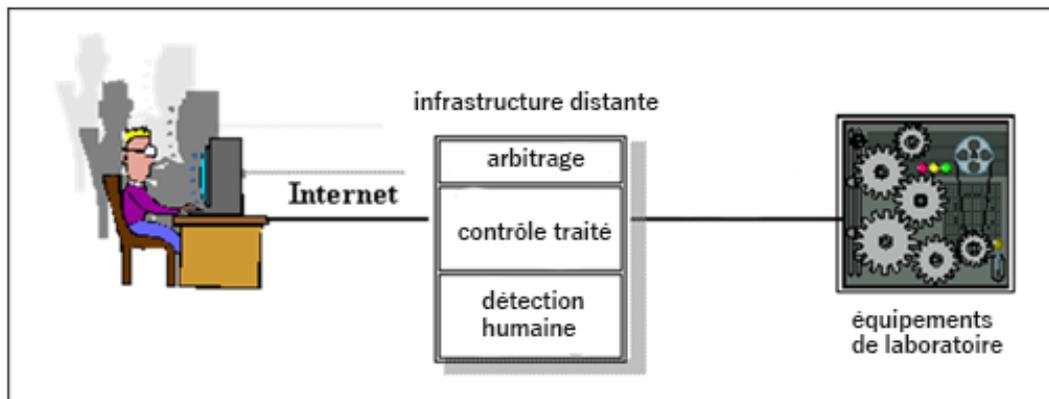


Figure I.6 : Principe d'un laboratoire distant

Du côté de l'utilisateur, l'infrastructure distante remplit les fonctions clés suivantes:

- Fournit une interface utilisateur qui permet à l'appareil de l'expérimentation d'être surveillé et utilisé.

- Gère l'accès au laboratoire pour garantir qu'un seul utilisateur peut utiliser une expérience à la fois.

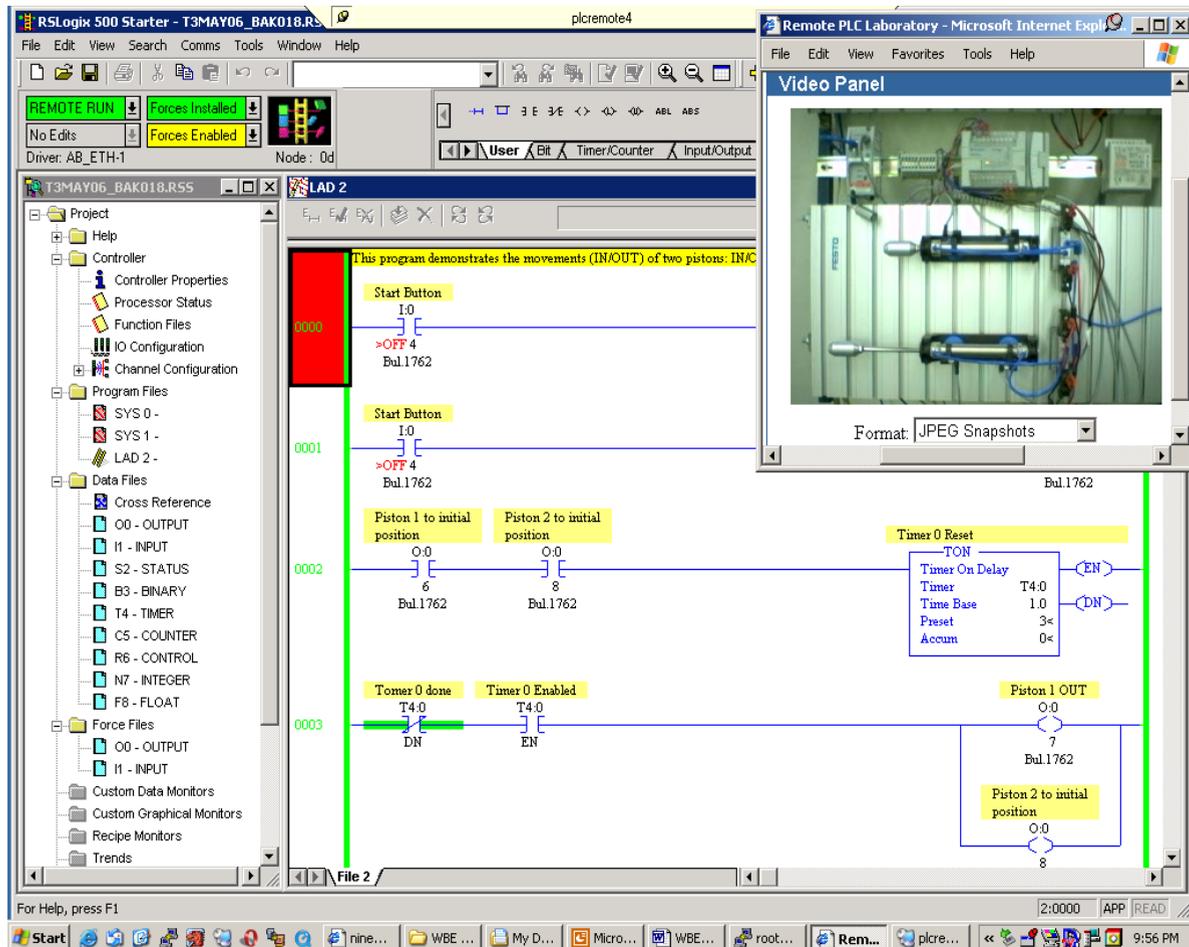


Figure I.7 :Exemple d'une interface utilisateur pour laboratoire distant

Côté équipement, l'infrastructure distante remplit les fonctions clés suivantes:

- Surveillerl'appareil, par exemple. grâce à l'utilisation de caméras vidéo, de microphones et d'autres capteurs.

- Contrôler l'appareil, par exemple. par l'utilisation d'interfaces d'E / S, de moteurs ou d'autres actionneurs

- Garantir que l'expérience est «nettoyée» à la fin de la session d'un utilisateur en réinitialisant automatiquement l'appareil ou en le mettant dans un état stable.

I.4.1 Laboratoires virtuels

Il existe plusieurs définitions des laboratoires virtuels dans la littérature. Selon (James, 1999), un laboratoire virtuel est défini comme : « un espace de travail électronique pour la collaboration à distance et l'expérimentation dans la recherche ou dans d'autres activités créatives, en vue de générer et de diffuser des résultats au moyen de technologies partagées de l'information et de la communication»[37]. Cette définition désigne beaucoup plus les laboratoires de recherche que les laboratoires scientifiques et techniques. [38]définissent les laboratoires virtuels comme étant des environnements informatiques qui simulent les expériences qui peuvent avoir lieu dans un laboratoire réel en utilisant la technologie de la réalité virtuelle. [39]caractérise les laboratoires virtuels par des simulations de scénarios expérimentaux modulaires conçus pour être mis en œuvre à partir d'un ou plusieurs ordinateurs. Les modèles mathématiques sont mis à contribution pour s'approcher le plus près de la crédibilité des simulations représentant des concepts théoriques ou des dispositifs réels. Dans certains cas, il n'est pas possible de simuler des scénarios expérimentaux ou le comportement de dispositifs réels. C'est le cas où les modèles mathématiques sont trop complexes, par manque de disponibilité de puissance de calcul ou quand les délais des traitements informatiques sont longs. Par conséquent, le recours vers les laboratoires réels à distance est une solution prometteuse[6].

I.4.2 Laboratoires hybrides

Il existe une autre classe de laboratoires visant à rassembler le meilleur des deux mondes (distants et virtuels) : il s'agit d'un laboratoire hybride. Ce laboratoire est en partie composé de manipulation sur des dispositifs distants et réels, mais il comporte des simulations, issues d'une étape de modélisation, comme pour les laboratoires virtuels.

I.5 L'apprentissage dans laboratoire d'ingénierie

Maintenant, l'enseignement est basé sur la recherche sur une idée constructiviste, que l'apprentissage est un processus par lequel l'étudiant construit ses propres idées qui sont reliées à d'autres idées dans des réseaux de plus en plus complexes. Le modèle constructiviste, lorsqu'il est pratiqué, est un changement radical par rapport aux pratiques traditionnelles d'enseignement et d'apprentissage. Les enseignants sont souvent mal informés de ces nouveaux modèles d'apprentissage [7,8] et de leurs implications pour l'enseignement en classe et dans l'élaboration de curriculum. Les comportements en classe

de nombreux enseignants continuent de suggérer la croyance traditionnelle selon laquelle les connaissances sont directement transmises aux bons étudiants et qu'il faut s'en souvenir comme transmis. Dans les sciences dites expérimentales, le laboratoire est le lieu dans lequel les étudiants développent leur compréhension des concepts scientifiques, des compétences de recherche scientifique et des perceptions de la science. Le laboratoire de sciences est un environnement d'apprentissage unique aussi, il est un cadre dans lequel les étudiants peuvent travailler en petits groupes pour étudier des phénomènes scientifiques.

Les activités de laboratoire ont le potentiel d'améliorer les relations constructives sociales ainsi que les attitudes positives et la croissance cognitive [9,10]. L'environnement social dans un laboratoire est généralement moins formel que dans une salle de classe conventionnelle ainsi, le laboratoire offre des possibilités d'interactions productives et coopératives entre les étudiants et avec l'enseignant qui ont le potentiel de promouvoir un environnement d'apprentissage particulièrement positif. L'environnement d'apprentissage dépend beaucoup de la nature des activités menées dans le laboratoire, des attentes de l'enseignant (et des étudiants) et de la nature de l'évaluation. Ceci est partiellement affecté par les matériaux, les équipements, ressources et environnement physique, mais l'environnement d'apprentissage qui produit est beaucoup plus que la fonction et les attentes de l'apprentissage, la coopération et les interactions sociales entre les étudiants et les enseignants, la nature de la recherche menée dans le laboratoire [11]

I.6 Impact socio-économique

La révolution des technologies de l'information et de la communication nous a inspiré aujourd'hui que le monde entier a changé et que les méthodes scientifiques et professionnelles ont changé avec lui, transformant l'espace virtuel en une réalité tangible dans divers domaines et spécialités vitales des économies nationales. L'enseignement à distance, ou en particulier les laboratoires à distance, a contribué à résoudre de nombreux problèmes économiques pour les universités. Nous essaierons de résumer cela en quelques points.

- Résoudre le problème du manque d'équipement dans les laboratoires réels

De nombreuses universités ont des problèmes économiques. Cela conduit à une grave pénurie de matériel de laboratoire en fonction du nombre d'étudiants. Les laboratoires à distance ne nécessitent pas beaucoup d'équipements, un seul appareil peut être utilisé pour tous les étudiants. Cela contribue à réduire les factures de laboratoire.

-Réduire la main-d'œuvre

Dans les laboratoires à distance, vous n'avez pas besoin de beaucoup de techniciens, vous n'avez besoin que d'un seul technicien pour équiper les outils du laboratoire réel pour une fois, ensuite, tout se fait automatiquement.

-Résoudre problèmes l'hébergement et transport

Pendant l'enseignement à distance, vous n'avez pas besoin de fournir l'hébergement et le transport des étudiants, tout se fait à la maison.

I.7 Conclusion

Dans ce présent chapitre, nous avons passé en revue des notions sur les systèmes embarqués et leur émergence dans tous les domaines de notre vie actuelle, notamment dans notre les solutions pilotables à distance et particulièrement les labos à distance. Nous sommes donc aujourd'hui entourés, on pourrait presque dire envahis, par les systèmes embarqués puisqu'ils sont présents absolument partout. La conception de ces systèmes pose de nombreuses contraintes spécifiques : consommation d'énergie, encombrement, parcimonie des ressources matérielles disponibles, etc. Quel que soit le contexte de l'application embarquée à développer, les logiciels libres offrent des solutions viables aussi bien techniquement qu'économiquement. Dans le chapitre suivant, nous détaillerons les différentes architectures utilisées dans les laboratoires distants.

Chapitre II

Laboratoires distants : projets, caractéristiques matérielles et logicielles

II.1 Introduction

Des plates-formes d'enseignement à distance ont été développées spécialement pour faciliter l'apprentissage et suivre l'enseignement et la pédagogie des apprenants. En sciences techniques, l'autre défi face aux moyens d'enseignement à distance, ce sont les travaux pratiques. Il faut reproduire les tâches effectuées par les étudiants en laboratoire mais à distance. Loin d'être un TP virtuel, l'apprenant doit manipuler à distance du matériel, et doit faire des mesures à distance, ce qu'on peut appeler TP à distance ou télé-TP.

Dans ce chapitre, nous détaillerons différentes solutions de TP distants, proposées dans la littérature.

A la fin de ce chapitre, nous donnons une description globale de notre contribution sur les laboratoires distants que nous avons nommé projet **LABORATOIRE DE TECHNOLOGIE EL BAYADH (LABTEC)**.

II.2 Projet iLAB

iLabs est dédié à la proposition selon laquelle les laboratoires en ligne - de véritables laboratoires accessibles via Internet - peuvent enrichir l'enseignement des sciences et de l'ingénierie en élargissant considérablement la gamme d'expériences auxquelles les étudiants sont exposés au cours de leur formation. Contrairement aux laboratoires conventionnels, les iLabs peuvent être partagés dans une université ou dans le monde entier. La vision d'iLabs est de partager aussi largement que possible les équipements coûteux et le matériel pédagogique associés aux expériences en laboratoire dans l'enseignement supérieur et au-delà.

Les équipes iLab ont créé des laboratoires à distance au MIT dans les domaines de la microélectronique, du génie chimique, de la cristallisation des polymères, de l'ingénierie structurelle et du traitement du signal comme études de cas pour comprendre les exigences complexes de l'exploitation d'expériences de laboratoire à distance et étendre leur utilisation à de grands groupes d'étudiants au MIT et autour du monde.

Sur la base des expériences des différentes équipes de développement iLab, le projet iLabs développe une suite d'outils logiciels qui rend efficace la mise en ligne et la gestion d'expériences complexes en laboratoire. L'architecture partagée iLabs a les objectifs de conception suivants:

- Minimisez les efforts de développement et de gestion pour les utilisateurs et les fournisseurs de laboratoires distants.
- Fournir un ensemble commun de services et d'outils de développement.
- Échelle à un grand nombre d'utilisateurs dans le monde entier.
- Permettez à plusieurs universités dotées d'infrastructures réseau diverses de partager l'accès.

II.2.1 Partie logicielle

La coté logicielle iLab est en mesure de fournir des logiciels contextuels et spécialisés pour devenir un laboratoire en ligne. Ses principales caractéristiques sont les suivantes

- systèmes d'exploitation différents pour développer des expériences sur plate-forme
- permet le partage d'expériences entre différentes universités ;
- fournit des outils de gestion efficaces pour les fournisseurs de laboratoire ;
- à une conception évolutive.

L'architecture globale nommée iLabShared Architecture (ISA) [12], [13] schématisée dans la figure II.1, se décompose en trois modules connectés par des services Web :

- Le serveur de laboratoire : il contrôle la partie matérielle et est géré par le fournisseur du laboratoire ;
- Le client qui tourne sur le poste de l'utilisateur distant et fournit l'interface graphique de manipulation ;
- Le courtier de service (Service Broker) sert d'intermédiaire entre le serveur et le client et fournit les services de stockage, d'administration. Il peut être commun à plusieurs laboratoires d'une même université.[40]

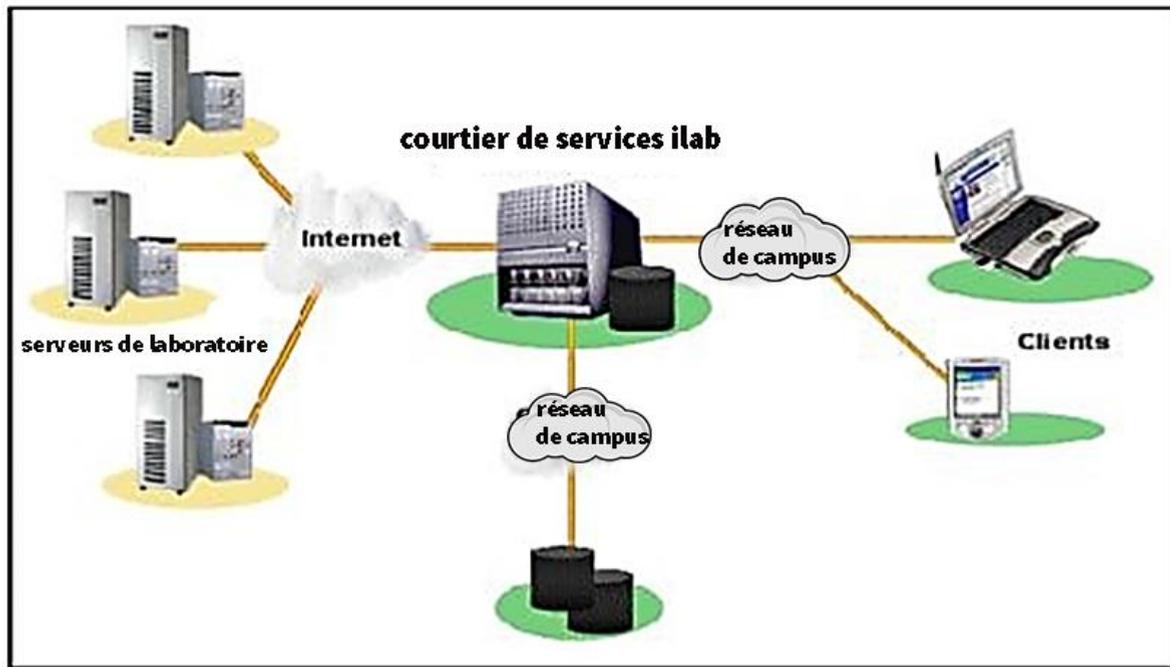


Figure II.1 : Architecture d'iLab [14].

II.2.2 Partie matérielle

Afin de contrôler des instruments de mesure et la matrice de commutation dans iLab, on a utilisé le serveur qui dispose d'une carte d'interface Agilent GPIB. Les services Web doivent aussi rendre plus facile à intégrer dans l'architecture globale les modules fournis par les fournisseurs. La matrice de commutation permet au système d'accueillir plusieurs dispositifs semi-conducteurs pour la caractérisation.

Le projet iLab a mis l'accent sur la construction des laboratoires distants à base de la plate-forme NI-ELVIS(National Instrument EducationalLaboratoryVirtual Instrumentation Suite) [15]. Parmi les instruments disponibles sur NI-ELVIS : un générateur de fonctions, un oscilloscope, un multimètre, ...etc. Ces instruments permettent aux étudiants d'effectuer des mesures en temps réel sur les circuits électroniques et faire des tests et débogage sur des circuits analogiques et numériques. L'utilisateur peut modifier la configuration du système en cours de test grâce à une matrice de commutation [13].[40]

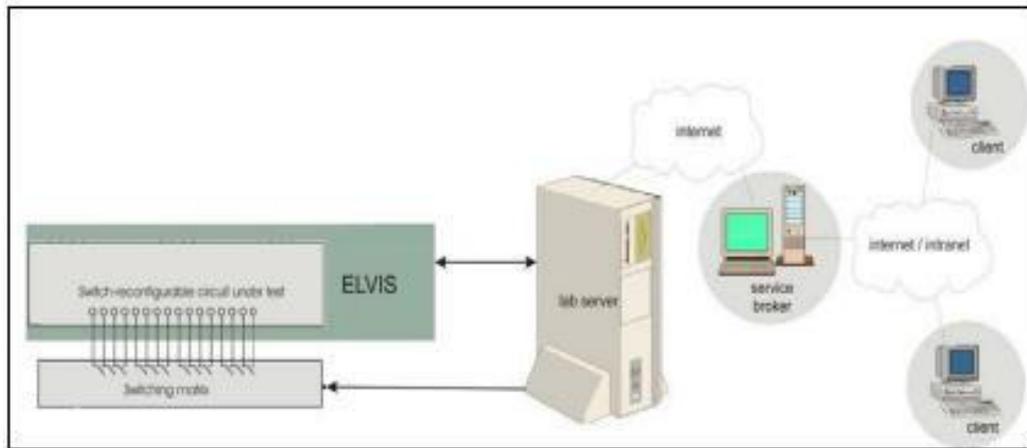


Figure II.2 : Architecture d'iLab basée sur plateforme ELVIS [13].

II.3 Projet VISIR

VISIR est un laboratoire à distance pour les circuits électriques et électroniques. C'est un "vrai" laboratoire avec des proto-cartes, des sources, des générateurs de signaux, des compteurs, des oscilloscopes et des composants (résistances, condensateurs, inductances, diodes, etc.). De plus, il a une caractéristique très importante - il peut être commandé à distance via Internet. Cette caractéristique permet aux utilisateurs du monde entier d'utiliser le laboratoire 24h / 24 et 7j / 7. C'est un puissant outil de coopération entre les institutions.

L'institut de technologie « Blekinge Institute of Technology » (BTH) en Suède a développé un projet appelé VISIR (Virtual Instrument System in Reality), conjointement avec National Instruments aux états unis comme fournisseur d'équipement de mesure et Axiome duTech en Suède comme fournisseur de solutions éducatives, techniques et logicielles. Ce projet a diffusé un concept de deux laboratoires en ligne, créé à BTH en utilisant des techniques open sources, en collaboration avec d'autres universités et organisations. Le premier laboratoire concerne les expériences de mesures électriques. Le deuxième laboratoire concerne des expériences mécaniques de vibration. Le concept est basé sur l'ajout d'une commande à distance des laboratoires à l'enseignement classique, pour que ces derniers soient plus accessibles.[11] , [40]

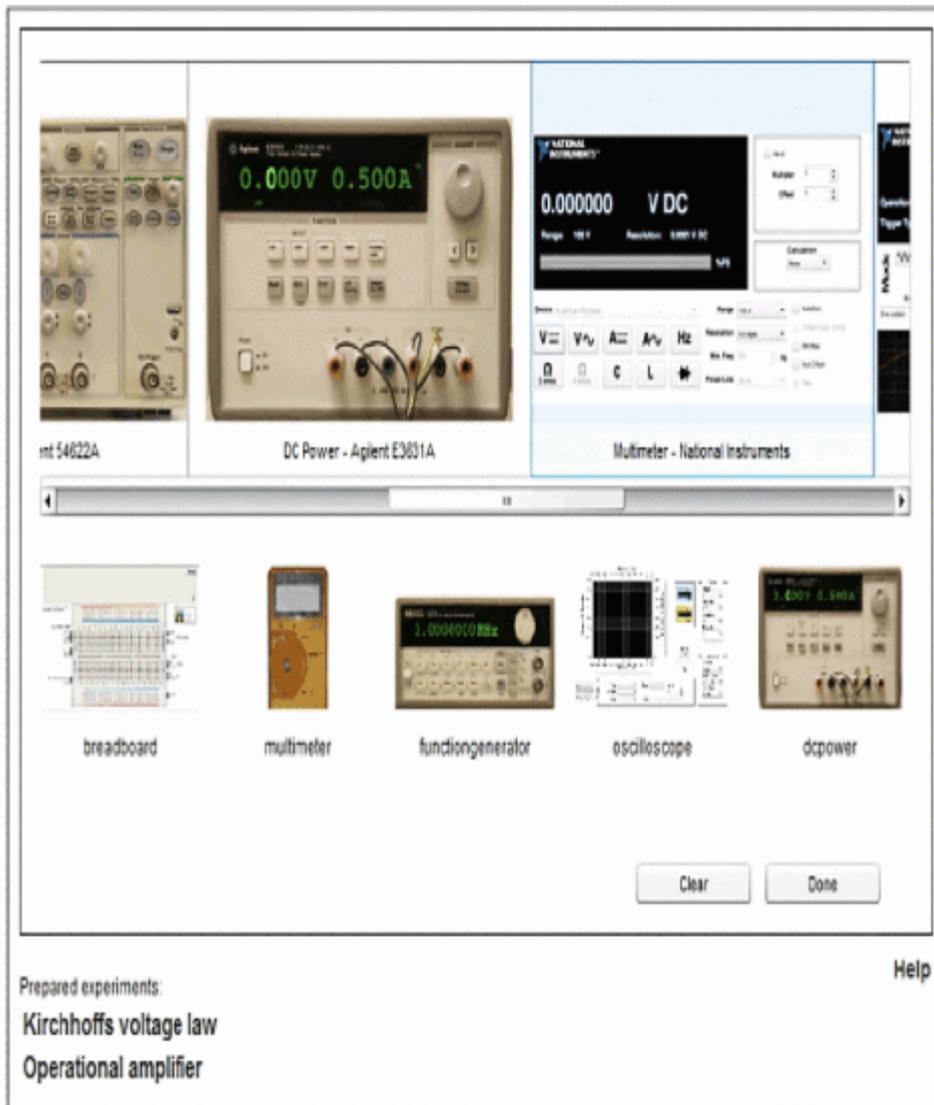


Figure II.3 : L'étudiant sélectionner l'interface instrument avec laquelle il est familier. des instruments utilisés dans VISIR [40].

II.3.1 Partie logicielle

La plate-forme logicielle utilisée pour développer le laboratoire VISIR est représentée par la Figure II.4. Le projet VISIR est open source, la documentation et les codes source sont accessibles à partir de l'url <http://openlabs.bth.se/>..[40]

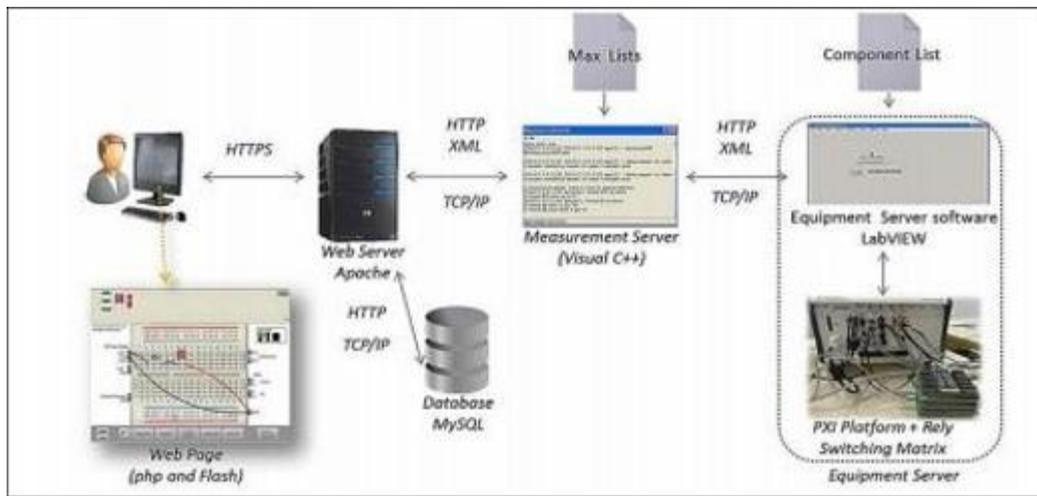


Figure II.4 :Architecture du projet VISIR [40].

La plate-forme met en œuvre différents serveurs :

- Le serveur web assure le lien entre l'étudiant distant et l'expérience située à l'université.
- Le serveur de mesure a pour rôle principal de répondre aux requêtes de mesure envoyées par les clients. Ces requêtes sont codées en utilisant un protocole d'expérimentation contenant les paramètres et les fonctions de l'instrument utilisé dans l'expérience. Les requêtes sont ensuite validées et transformées dans un format compréhensible par les serveurs de l'instrument directement connecté à l'équipement. Ce serveur utilise Microsoft Windows comme système d'exploitation. Le programme est écrit en C ++ en utilisant Microsoft Visual C ++.
- Le serveur des équipements est chargé de recevoir les requêtes de serveur de mesure et configurer les instruments et la matrice de commutation, selon les demandes. Il a été développé en Labview de National Instruments [12].

L'interface utilisateur utilisée pour créer et tester le circuit a été développée en flash dans un premier temps puis en HTML/JavaScript[32][17].

L'authentification et les files d'attente sont également traitées. L'interface entre le serveur de mesure et le serveur de l'équipement utilise le protocole TCP/IP, de sorte qu'ils peuvent être en cours d'exécution sur des machines distinctes.[40]

II.3.2 Partie matérielle

Le projet VISIR donne la possibilité de câblage du circuit à distance en utilisant une maquette virtuelle associée à système matricielle de relais de commutation. Il vise à d'appliquer une norme internationale, ce qui permet à des équipes à travers le monde

d'améliorer et de développer cette approche puissante dans les laboratoires en ligne, en utilisant un logiciel standard comme IVI (Interchangeable Virtual Instruments), des plates-formes d'équipements tels que PXI (PCI eXtensionsfor Instrumentation) et LXI (LAN Extensions for Instrumentation) [12].

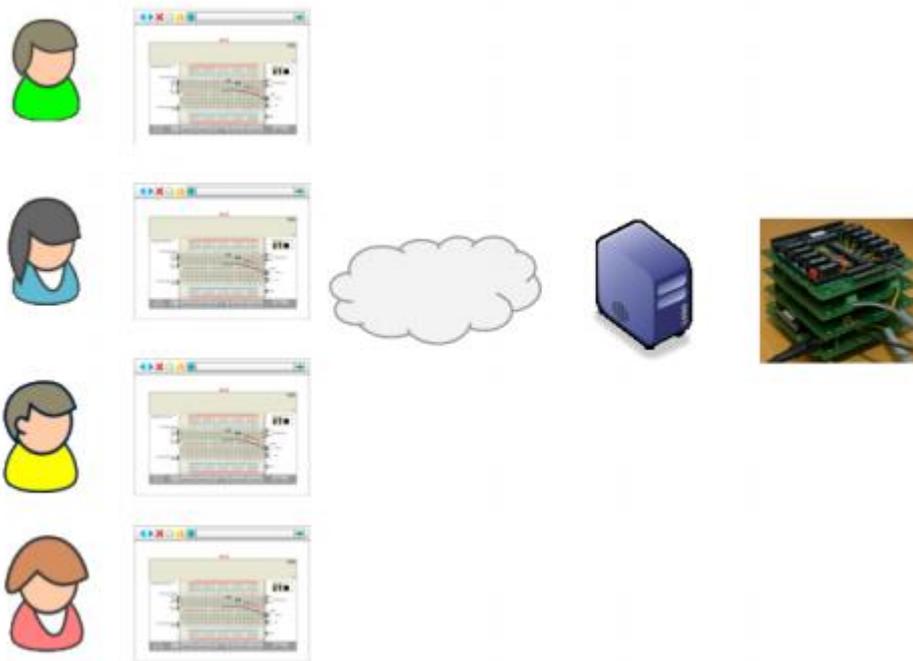


Figure II.5 : Exemple SPRL complexe: VISIR, prenant en charge plusieurs étudiants simultanés.[12]

Le laboratoire à distance VISIR utilise le National Instruments PXI ou la solution PCI pour rassembler tous les instruments nécessaires pour configurer et effectuer des mesures sur un circuit construit par les étudiants à distance. Les instruments disponibles sont: oscilloscope, générateur de fonctions, multimètre et alimentation. Tous ces instruments devraient être de National Instruments, parce que le serveur d'équipement fonctionne avec ces pilotes. Pour que l'utilisateur puisse construire un vrai circuit via l'interface Web, VISIR utilise une matrice de commutation propriétaire (**figure II.6**) commandée par le serveur d'équipement. Les relais sont disposés dans un modèle de matrice en trois dimensions avec des connecteurs pour les instruments et des prises pour les composants. Il y a deux modèles de cartes différentes : l'une pour les instruments et l'autre pour les composants. De cette façon, les commutateurs de relais sont intégrés dans le circuit par des fils d'une longueur limitée pour gagner en bande passante. Les nœuds de la matrice de

commutation sont propagés d'un bord à l'autre par un bus de nœud. La notation "nœud" se réfère au fait que chaque conducteur créée par ces connecteurs empilés peut être un nœud dans un circuit souhaité[13] , [40]

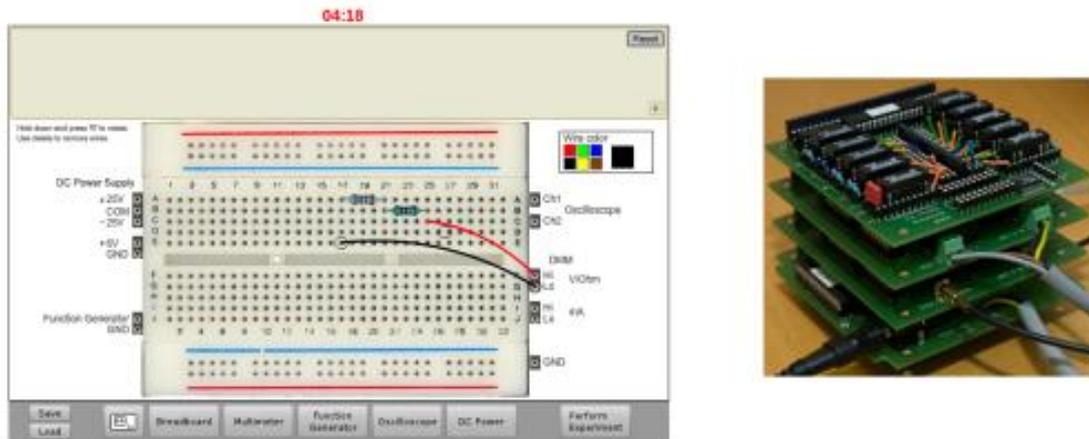


Figure II.6 :Projet VISIR: carte électronique utilisateur (à gauche) et matrice de commutation (droite) [16].

Dans une session de VISIR, l'étudiant peut créer le circuit à distance par un navigateur web tandis que le serveur de mesure et le serveur d'équipement sont chargés de réaliser ce circuit sur la matrice de commutation.

II.4 Projet ISILAB

ISILab qui signifie Internet Shared Laboratoire d'instrumentation, est une télécommande électronique Laboratoire développé à l'Université de Gênes. C'est aggravé par plusieurs expériences indépendantes qui ont été développés sur ISILab. Toutes ces expériences ont le même calendrier schéma, basé sur les files d'attente. Comme dans VISIR ou chez l'utilisateur entre dans une session mais n'acquiert pas l'expérience. Au cours de cette session, l'utilisateur soumettra des configurations qui sera mis en file d'attente et exécuté dans les expériences physiques. Ces soumissions prennent généralement entre 500 et 700 millisecondes, mais pourrait atteindre les 2 secondes en une seule expérience. Le schéma d'ordonnancement utilisé est donc une file d'attente. Au cours du cours 2009-2010, le système a été utilisé entre 800 et 1000 fois par 30 à 40 étudiants, en utilisant tous les expériences énumérées ci-dessus. La file d'attente a atteint files d'attente de moins de 5 utilisateurs[40].

II.4.1 Partie Logicielle

Pour interagir avec les expériences, l'utilisateur n'a besoin ni de matériel ni de logiciel spécifique sauf un navigateur web et la machine virtuelle Java. À travers les panneaux des instruments virtuels représentés sur le site web, l'utilisateur peut commander à distance ces instruments réels et effectuer des expériences.

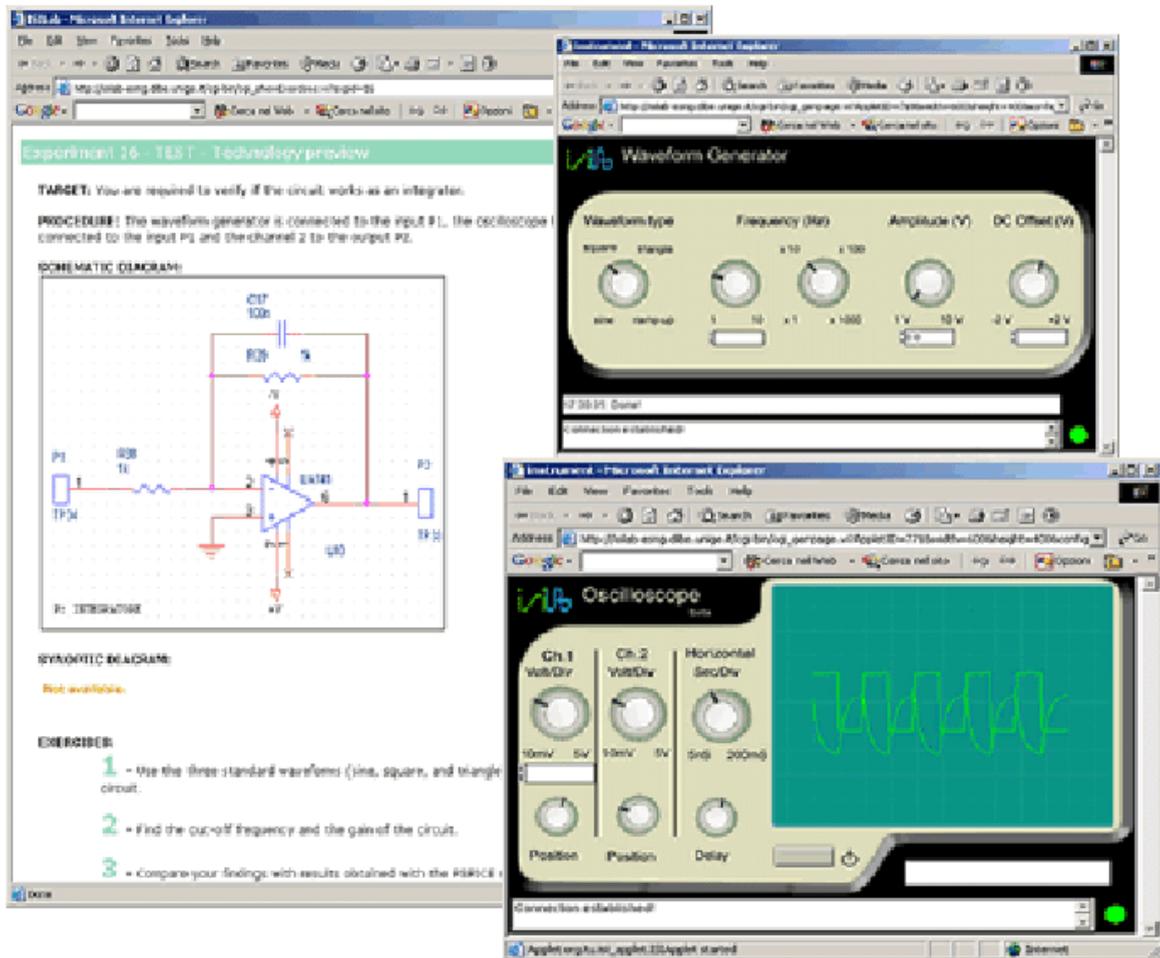


Figure II.7 : Interfaces utilisateur d'ISILab [40]

Ces panneaux virtuels sont définis selon un schéma XML spécifique et les interfaces graphiques d'utilisateur (GUI) peuvent être configurées en fonction des objectifs de chaque expérience. ISILab possède des modules configurables en apparence et comportement qui peuvent être utilisés par la suite pour contrôler les différents instruments. Cette approche est basée sur un logiciel appelé Applet VIEW. Cet environnement permet au développeur de construire une applet intégrée avec des programmes écrits sous Labview. Les fichiers de

configuration d'AppletVIEW sont écrits avec un langage basé sur XML appelé VIML (Virtual Instrument Mark-upLanguage)[42]. ISILab a développé ces méthodes pour donner une applet Java qui fonctionne avec ces fichiers de configuration XML[12][20][40].

II.4.2 Partie matérielle

ISILab supporte à la fois les instruments autonomes et ceux connectés sur ordinateur (PCI, PXI, GPIB, etc.). Il rend transparent le matériel à l'aide d'une interface de programme d'application homogène dans un serveur de laboratoire [20]. De cette façon, si un nouvel instrument va être ajouté, il doit seulement être enveloppé dans un adaptateur de pilote approprié pour exposer les fonctionnalités de l'appareil. Ces adaptateurs pilotes sont basés sur la technologie IVI (Interchangeable Virtual Instruments) et ils sont développés pour l'oscilloscope, générateur de fonctions et le multimètre numérique.

Si les étudiants veulent créer un circuit à l'aide de composants discrets, le nombre de composants et de relais pour les interconnecter augmente de façon exponentielle avec le nombre de composants. Pour cette raison ou pour d'autres, ISIBoard a été créé. Il s'agit d'une carte mère avec seize slots, où se placent les cartes expériences (Figure II-8). Chaque carte a dix-huit lignes pour fournir l'alimentation, les signaux d'entrée/sortie et les connexions avec les instruments. Lorsque le circuit est monté sur la carte mère, les connexions avec les instruments sont créées dynamiquement par un ensemble de commutateurs. Autrement dit, cette carte mère fonctionne comme une matrice de commutation commandée par le serveur de laboratoire, chaque carte possède un identificateur unique et un tableau configurable qui permet à chaque circuit de partager la même instrumentation. En fonction de l'expérience sélectionnée par l'utilisateur sur le portail Web, la carte correspondante va être connectée aux instruments et sur le portail [19].[40]



Figure II.8 : Carte mère (ISIboard) avec les cartes d'expériences [40].

II.5 Projet NETLAB

NetLab (<http://netlab.unisa.edu.au/>) est un laboratoire à distance développé par l'Université du Sud de l'Australie depuis 2002. Il est actuellement utilisé par les étudiants de l'École de génie électrique pour les travaux pratiques en dehors des heures prévues. NetLab est un environnement interactif et collaboratif multiutilisateur, il donne un accès distant via Internet à des vrais instruments et des circuits pour effectuer des mesures en temps réel. Grâce à son interface web graphique, l'utilisateur avec un constructeur de circuit a la possibilité de créer de véritables expériences et de les manipuler en utilisant des touches et des boutons intégrés dans cette interface [21] [22]. La Figure II-9 montre l'architecture de base de NetLab.[40]

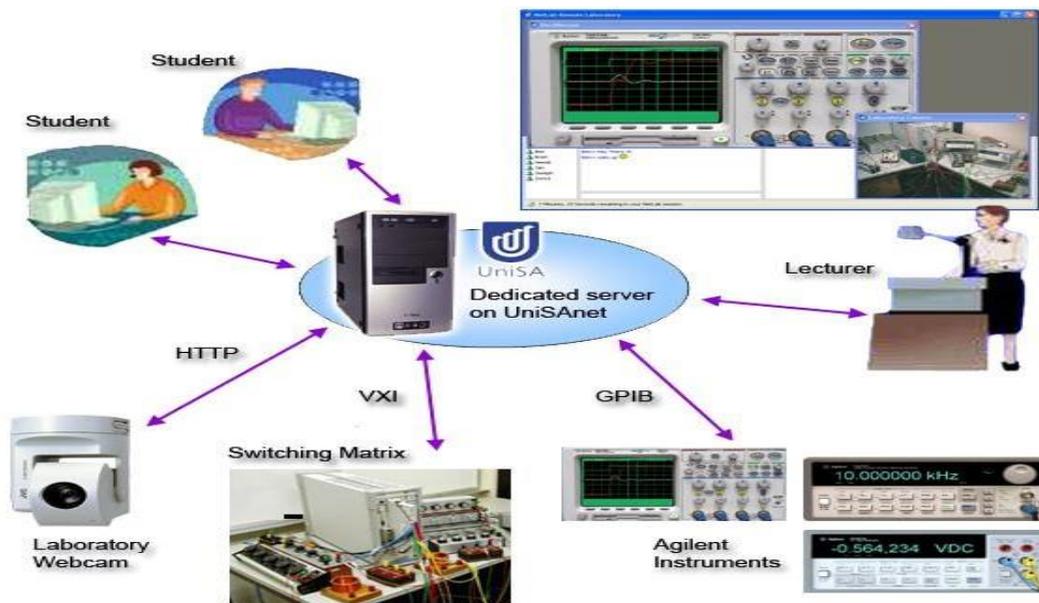


Figure II.9 : Architecture de NETLAB[22]

Un système réel situé dans un laboratoire est connecté au serveur NetLab via GPIB (General Purpose Interface Bus). Le serveur est connecté à Internet. Les appareils réels sont ensuite contrôlés à distance sur Internet par un utilisateur à partir d'un emplacement distant.

II.5.1 Partie logicielle

NetLab est une application développée en Java. Elle contient les interfaces des instruments dessinées avec le logiciel Java 2D drawing. Elle fonctionne sur n'importe quel système d'exploitation à condition qu'il dispose d'une installation de l'environnement d'exécution Java SE, version 3.0 ou ultérieure. Pour exécuter l'application NetLab, les étudiants doivent télécharger et installer une applet en créant un raccourci sur le bureau. L'accès à NetLab est libre, [22].

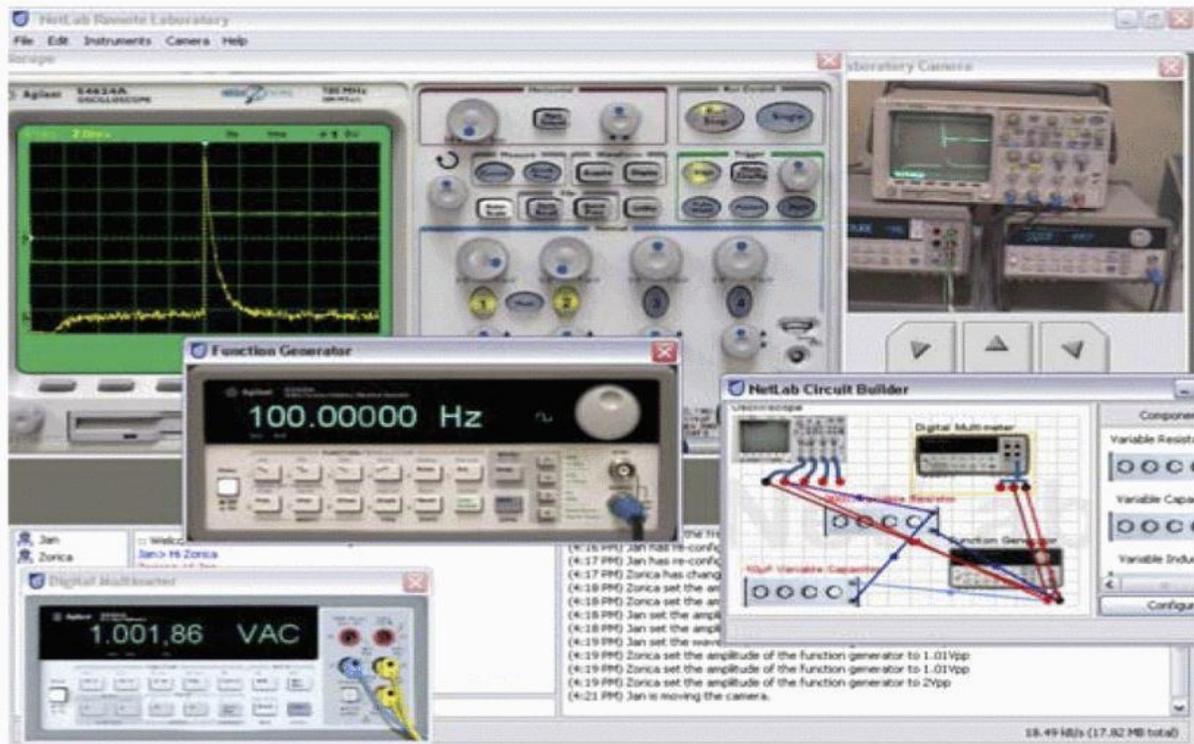


Figure II.10: Les interfaces des instruments utilisés dans NETLAB[22,24]

II.5.2 Partie matérielle

Le laboratoire dispose des instruments suivants : un oscilloscope, un générateur de signaux, un multimètre, et une matrice de commutation. Ils peuvent être tous contrôlés en ligne. L'utilisateur peut télécharger les données de l'expérience à distance sur son PC pour un traitement ultérieure [22] [23]. La figure II.11 La plate-forme physique de NetLab.

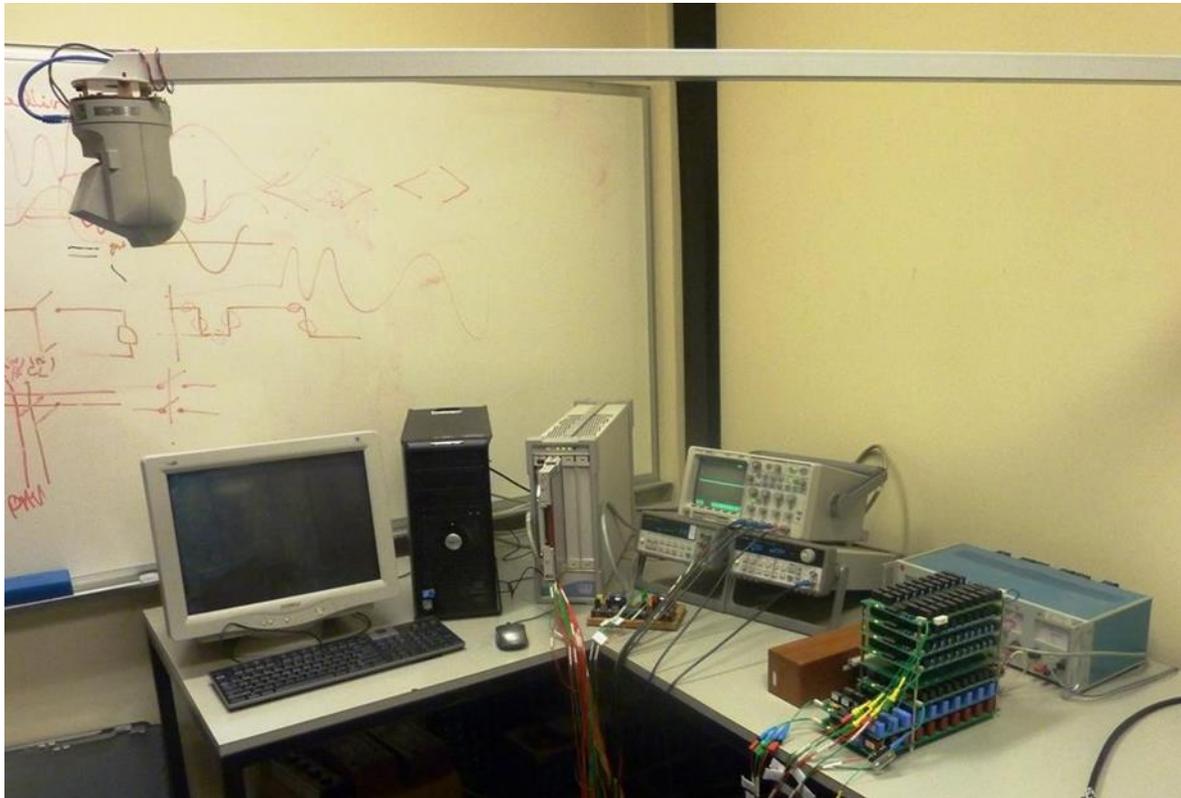


Figure II.11: plate-forme physique de NetLab[23]

Le générateur de circuit de NetLab est basé sur une matrice de commutation Agilent E1465A avec 16x16 modules disponibles. Cette matrice de commutation de relais nécessite un matériel qui comprend: une unité centrale E8408A VXI avec 4-slots et un module de commande E1406A. Ces composants communiquent avec le serveur qui héberge l'application de NetLab par l'interface GPIB, tandis que le protocole de communication standard VXI est utilisé pour la communication interne dans le module de commande [40].

La figure II.12 montre l'interface graphique pour générer un circuit dans le laboratoire NetLab.

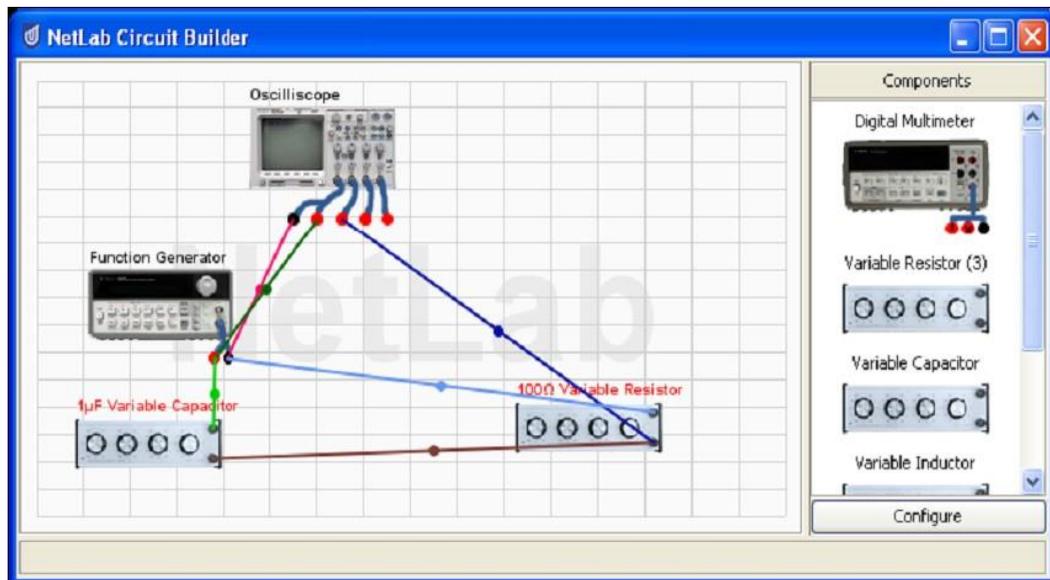


Figure II.12 : Interface de générateur de circuit de NetLab[23].

II.6 Projet RwmLAB

Dans ce projet, une nouvelle architecture appelée "RemoteWiring and Measurement Laboratory (RwmLab)" est proposée pour permettre aux étudiants de câbler physiquement les appareils électriques et électroniques circuits utilisant un accès Internet. De même, ils pourront prendre des mesures réelles sur leur accès Internet. De cette façon, les étudiants connaîtront les écrous et les volts, les frustrations et expérience pratique d'un environnement de laboratoire réel tout en accédant au laboratoire à distance. L'interface de RwmLab sera grandement simplifiée en utilisant un câblage graphique environnement. Cependant, le système câblera physiquement les composants ensemble au laboratoire hôte site.

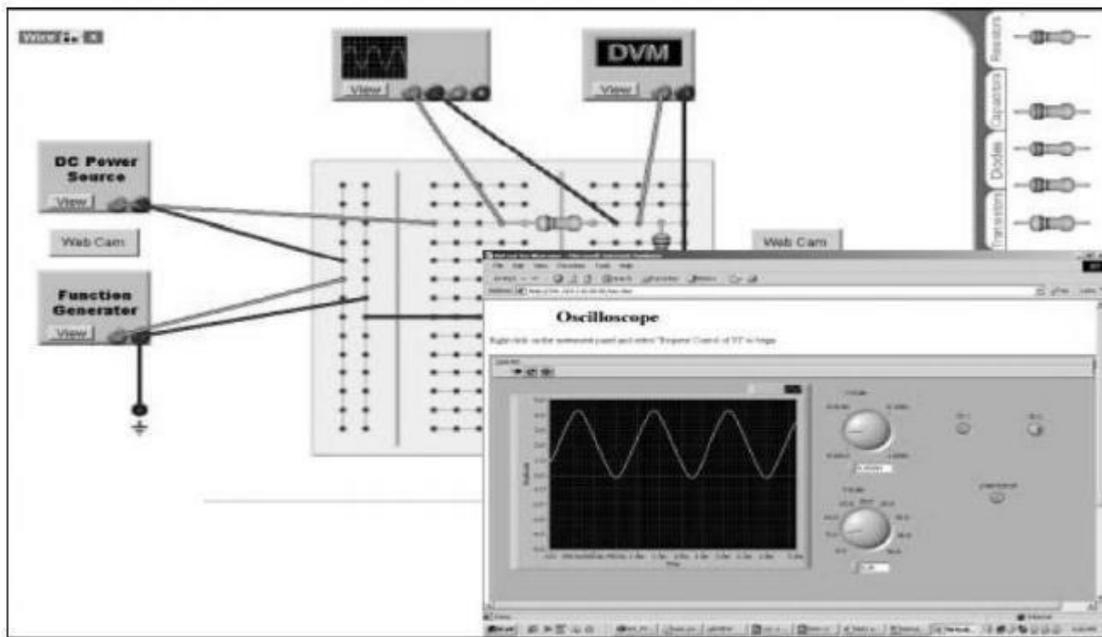


Figure II.13: Interface utilisateur de RwmLab [25]

II.6.1 Partie logicielle

RwmLAB est développé en HTML et JavaScript. Lorsque l'utilisateur câble le circuit sur la carte de test virtuel, le logiciel transmet un code numérique qui représente les connexions des nœuds du circuit. Ce code est transmis en utilisant un protocole d'interface de grille commune (CGI) écrit en langage C. Les commandes et les affichages des instruments sont développés en utilisant le logiciel Labview de National Instruments [25][26].

II.6.2 Partie matérielle

Le laboratoire RwmLAB est doté des instruments suivants: un multimètre, un oscilloscope et un générateur de fonctions. Le câblage et les mesures dans ce laboratoire à distance s'effectuent à l'aide d'une carte de matrice de commutation Xecom AWC83A qui joue le rôle d'un contrôleur de la plate-forme avec un serveur web et un CPLD (Complex Programmable LogicDevice). Ce dernier est utilisé pour contrôler la matrice de commutation. Le back-end de la matrice est un microcontrôleur AM186ES AMD 40 MHz avec SRAM et une mémoire Flash (Figure II-14) [25].

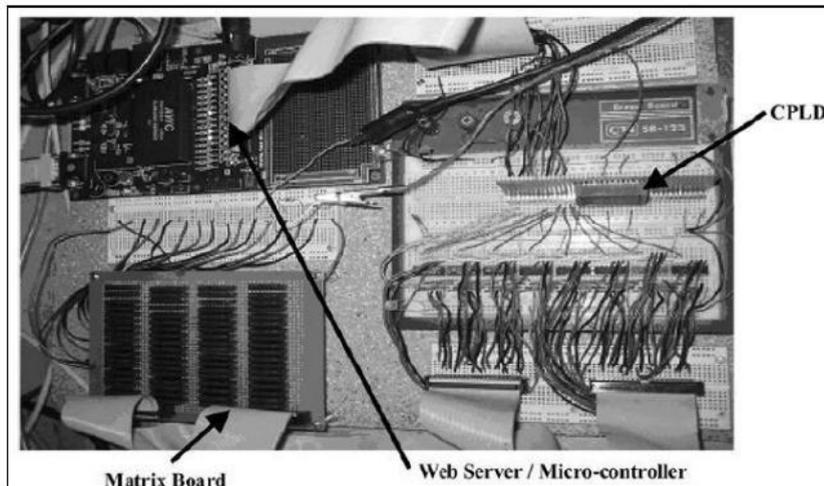


Figure II.14: La plateforme physique de RwmLAB[26].

Cette matrice de commutation comprend un tableau 8x8 configurable et basée sur des relais statiques, contrôlés par un serveur Web. Le circuit est défini par l'utilisateur à travers une interface visuelle représentant une carte de test virtuelle "Virtual breadboard", lui permettant de câbler physiquement un circuit électrique ou électronique. Les composants et les câbles placés autour de la carte virtuelle peuvent être déplacés pour accomplir la configuration souhaitée. Un ensemble d'instruments peut également être connecté à tous les nœuds du circuit. Lorsque l'utilisateur complète un circuit, le logiciel fait l'analyse pour déterminer les conducteurs qui sont reliés entre eux et configure les relais[25] [26].

II.7 Laboratoire distant LABTEC

Notre contribution consiste à développer un laboratoire distant par nos propres moyens disponibles dans nos laboratoires du centre universitaire, permettant de commander des instruments de mesure et de contrôler les circuits électriques à partir d'un navigateur Internet, dans l'enceinte du campus et à distance. Grâce à ce système, les étudiants auront la capacité de concevoir et de réaliser des expériences sans devoir être présents physiquement dans le laboratoire réel. Ainsi qu'un système complet qui permet aux enseignants et aux étudiants de réaliser de nombreuses tâches. L'enseignant peut ajouter un nouveau travail pratique et toutes les informations dont l'étudiant a besoin pour réaliser le travail pratique.

Une fois que l'étudiant a terminé son travail, l'enseignant peut évaluer son travail à travers le compte-rendu remis

Les étudiants peuvent interagir avec le circuit et des instruments de mesure via un navigateur Web.

Presque tous les laboratoires distants actuels utilisent un logiciel propriétaire (un logiciel semblable à Labview) et un matériel coûteux (un serveur) pour les mettre en œuvre. Dans notre projet, nous avons développé une solution en utilisant les moyens les plus simples possibles. Tels que Raspberry-Pi et les instruments de mesure universitaires. Le circuit utilisé dans notre exemple a été conçu pour redressement et filtrage monophasé avec un transformateur point milieu.

Une interface web est développée pour permettre à l'étudiant de configurer la partie matérielle des travaux pratiques en changeant par exemple de composants ou en positionnant le point de mesure en différents endroits du circuit étudié. Les circuits électriques sont dessinés à l'aide de l'outil SVG. l'ouverture et la fermeture des switch s'effectue en implémentant du javascript tandis que du côté serveur web nous utilisons l'environnement Django.

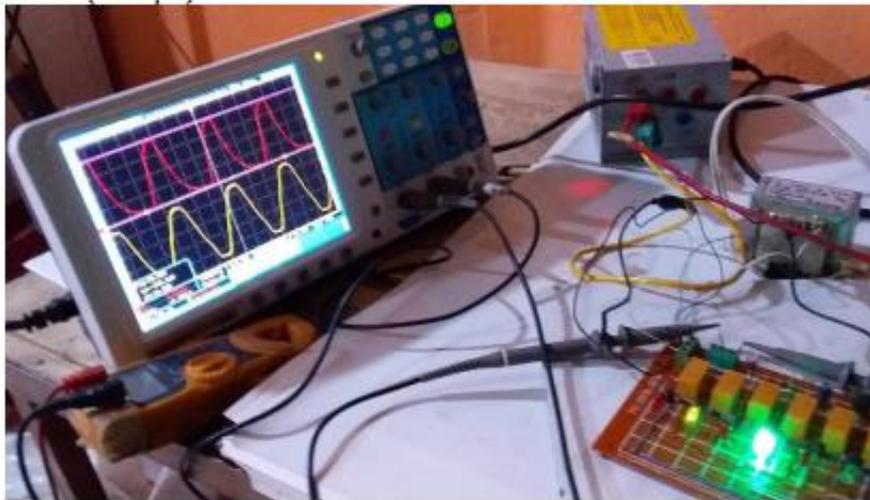


Figure II.15 : La plateforme physique de LABTEC.

II.7.1 Contraintes et caractéristiques du laboratoire LABTEC

A) Caractéristiques du laboratoire LABTEC

- Multi-utilisateur
- plateforme facile à manipuler par les utilisateurs
- Économique. Il n'a pas besoin d'appareils coûteux
- Open source. Et c'est évolutif

- permettre de développer et de réaliser des expériences sur des plate-forme et systèmes d'exploitation différents
- Utiliser un seule langage de programmation pour gérer le serveur et contrôler le GPIO (python) qui rend le système homogène et facile à maintenir
- L'utilisation des technologies web modernes (Ajax,Django,Flask ,Bootstrap ... etc). Facile et gratuit
- Permettre d'utiliser un oscilloscope disponible (avec ou sans serveur web intégré)

B) Contraintes du laboratoire LABTEC

- Bien que nous ayons fait de nombreuses facilités lors de l'ajout d'une nouvelle carte électronique de TP , mais doit y être ajouté code SVG . (Une fois seulement)
- Il n'est pas possible pour plus d'un étudiant d'accès laboratoire en même temps, nous avons donc ajouté un calendrier. Mais cela entraîne un problème qui prend beaucoup de temps à tous les étudiants pour terminer leurs travaux pratiques.

II.8 Conclusion

Dans ce chapitre, nous avons décrit différentes architectures de laboratoires distants qui ont été développés par la communauté scientifique, en expliquant leurs architectures matérielles et logicielles. Cette étude, nous a permis de nous situer par rapport à l'existant afin de proposer notre propre solution d'un laboratoire distant en implémentant des instruments de mesures disponibles et une architecture homogène et facile à implémenter par les enseignants et les étudiants.

Dans ce qui suit, nous expliquerons la partie conception et modélisation ainsi que le prototype réalisé dans le cadre de notre projet.

Chapitre III

Projet LABTEC, conception, réalisation, discussions des résultats

III.1 Introduction

Dans ce chapitre, nous détaillerons les différentes phases de réalisations de notre projet baptisé « LABTEC ». Nous allons décrire la conception et la modélisation du système par UML en ressortant les différents paquetages matériels et logiciels à mettre en œuvre. Puis la réalisation de l'architecture retenue en argumentant le choix des différents outils employés. A la fin, nous discuterons les résultats obtenus ainsi que les contraintes rencontrées lors de cette phase de réalisation.

III.2 Cahier des charges (spécifications fonctionnelles)

L'objectif principal de notre projet consiste à concevoir et développer une plateforme de laboratoire distant, tout en se basant sur la matériel de mesure existant dans notre laboratoire du centre universitaire. Après la discussion avec nos encadrants, nous allons résumer les différentes spécifications fonctionnelles et non-fonctionnelles relatives à notre projet. Le système se compose d'un serveur qui contrôle les instruments de mesure et paramétrer la carte électronique du TP. (Fermeture et ouverture des switch. Il contient également une base de données servant à stocker diverses informations nécessaires pour réaliser des travaux pratiques. Le serveur se connecte à Internet réseau du centre universitaire Les instruments de mesure sont connectés au serveur via le réseau local LAN. L'utilisateur (administrateur, enseignant ou étudiant) doit s'authentifier afin d'accéder aux différentes fonctionnalités qui lui sont offertes par le système. Les tâches principales de chaque utilisateur sont décrites dans la partie modélisation par la suite. Aussi, les fonctionnalités attendues du système sont :

a) Multi-utilisateur

Il existe trois types d'utilisateurs. L'enseignant. Étudiant. Administrateur. Chaque utilisateur a une interface spéciale et ils peuvent se connecter en même-temps aux systèmes

b) Similitudes avec laboratoire réelle

Le système doit contenir toutes les opérations que l'étudiant et l'enseignant effectuent dans laboratoire réelle, de la réalisation à l'évaluation et à la notation

c) Calendrier

Chaque TP doit être géré par un calendrier établi par l'enseignant et affiché aux étudiants concernés par ce TP. L'étudiant réalisera le TP durant la période horaire qui lui est réservée.

d) Notifications

En plus des notifications qui apparaissent sur l'interface pour les étudiants. Lorsqu'un nouveau travail pratique est ajouté par l'enseignant, un message est automatiquement envoyé à chaque étudiant dans un e-mail contenant les informations de TP et son calendrier

e) Information sécurisée

Le site contenant toutes les informations sur les enseignants et les étudiants, ces informations doivent être conservées et sécurisées.

f) Rapidité et flexibilité

Comme tout projet, l'application doit garantir une vitesse et une flexibilité optimales

g) Interface utilisateur/Expérience utilisateur (UI/UX)

L'interaction entre l'étudiant et l'interface doit être simple et rapide. De plus, le design doit être attractif et ergonomique pour éviter des anomalies de passage entre les différentes pages et web et permettre un acheminement homogène du processus du TP. En effet, ces spécifications devront être prises en considération dans n'importe quel projet à réaliser.

III.3 Conception et modélisation par UML

Après avoir consulté les enseignants encadrements et étudié le cahier des charges nous avons développé la conception finale du système, en se basant sur le modèle UML (Unified Modeling Language).

III.3.1 Conception UML

Pour décrire la conception finale de notre système, nous avons adopté le modèle UML (Unified Modeling Language) et ses différents diagrammes qui permettent de visualiser, spécifier, construire et documenter les abstractions d'un système logiciel. UML est considéré comme une boîte à outils qui offre des techniques de modélisation décrites par un langage.

III.3.1.1 Diagramme de cas d'utilisation

Pour décrire le comportement fonctionnel et les cas d'utilisation de notre système, on va utiliser les diagrammes de cas d'utilisations vu que cette notation graphique permet

de donner une vue générale simplifiée de notre système. Dans notre projet, nous avons identifié trois acteurs humains : Un enseignant concepteur et encadrant du Télé-TP, des étudiants qui réalisent le TP et l'administrateur de la plateforme. . Le rôle de chacun est comme suit :

Le diagramme des cas d'utilisation de notre projet est illustré par la figure III.1.

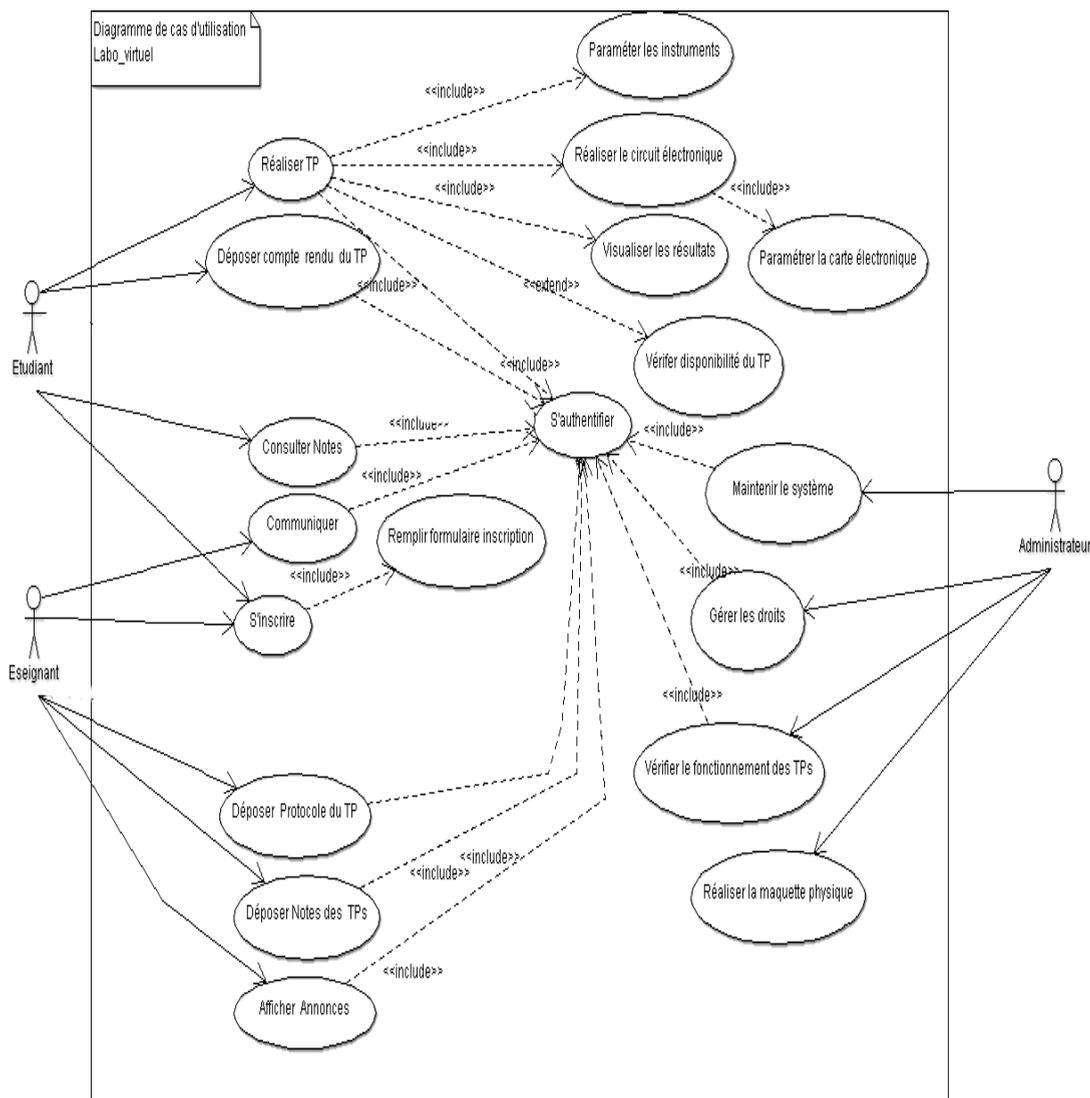


Figure III.1 : Diagramme de cas d'utilisation

Comme est décrit par la figure III.1, trois acteurs interagissent sur le système en effectuant divers tâches/ Ces acteurs devront s'inscrire au préalable avant d'effectuer leurs tâches. Le rôle de chaque acteur peut être résumé comme suit :

- **L'enseignant** : Prépare et dépose le protocole du TPs, corrige les compte-rendu et affiche les notes aux étudiants.

- **Etudiant** : Il s'agit de la personne effectuant le TP, dans le but d'atteindre des objectifs pédagogiques visés par ce TP.
- **Administrateur**: Il assure la maintenance du matériel utilisé, prépare la plateforme physique du TP et il gère les droits sur le système.

III.3.1.2 Diagramme de classe

Le diagramme de classe est un élément important dans une démarche de conception. Il nous permet de décrire les différents paquetages du système ainsi que les éléments qui seront développés. A travers les diagrammes de classes, nous pouvons énumérer les paquetages qui feront partie de l'architecture matérielle et ceux qui seront développés en logiciel.

En identifiant les concepts importants de notre projet, nous avons réalisé le diagramme de classes suivant pour représenter ces concepts et leurs associations.

La figure III.2, illustre les différentes classes qui seront développées ainsi que leurs interactions. Un utilisateur doit s'inscrire à travers un formulaire spécifique en remplissant différentes informations lui concernant. Une fois inscrit, pourra s'authentifier en utilisant son login et password et aura la possibilité de visualiser les différentes fonctionnalités par le système.

A noter qu'un TP est régi par un calendrier incluant la classe « date » permettant de paramétrer le planning. Il interagit aussi avec une classe « Carte_TP » qui est un composant matériel.

Dans un TP, l'étudiant manipulera des équipements et effectuera des mesures, d'où la relation entre la classe « TP » et la classe « Mesures ». Une fois avoir terminé son TP, l'étudiant aura à déposer un compte-rendu qui sera ensuite évalué par l'enseignant, d'où la classe « compte-rendu ».

Excepté la classe « carte_TP » qui sera réalisée par du matériel, les autres classes seront développées en logiciel.

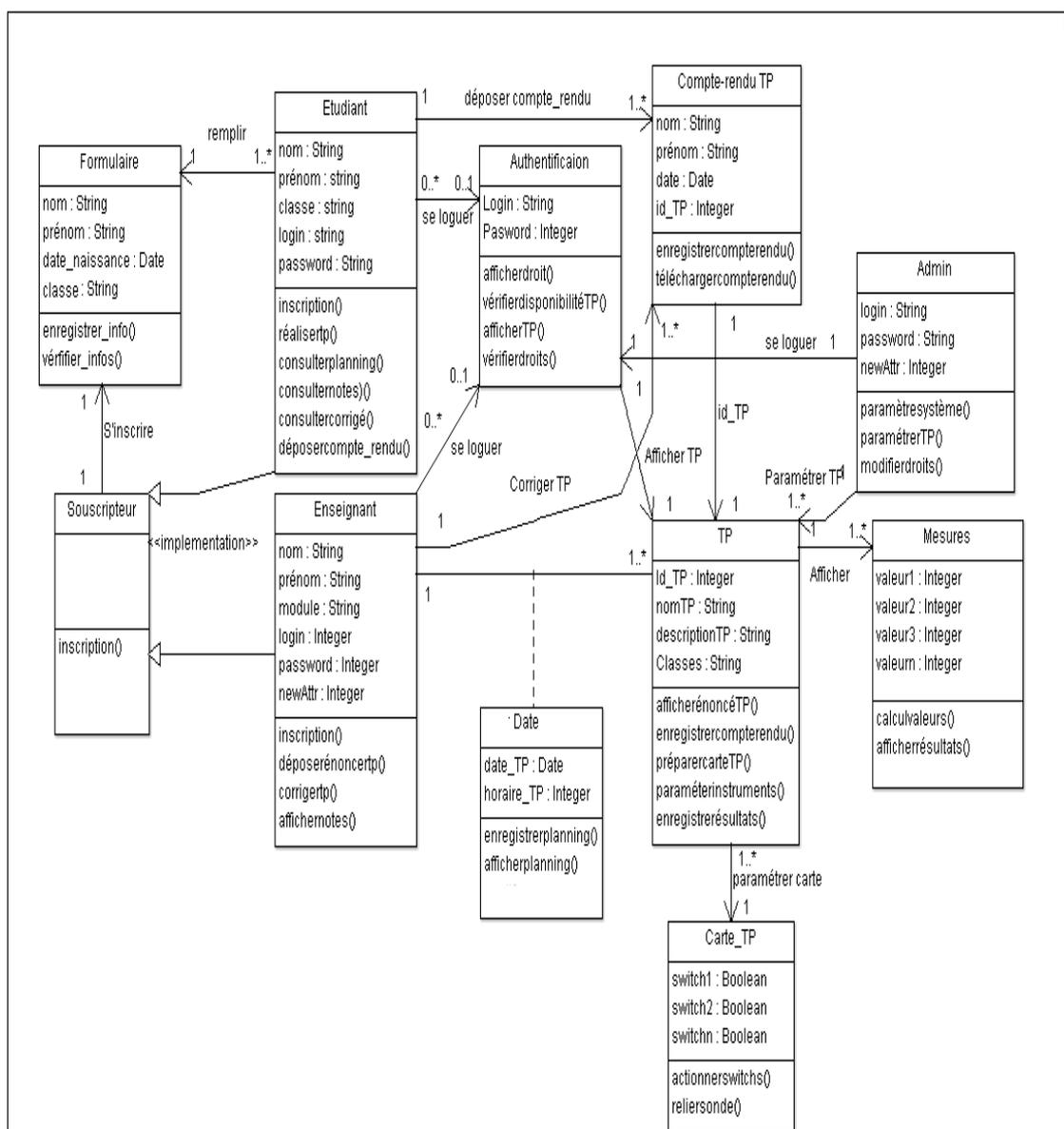


Figure III.2: Diagramme de classe de l'application

III.3.1.3 Diagramme de séquence

Le diagramme de séquence représente graphiquement les interactions entre les acteurs et le système selon un ordre chronologique. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets dans le cadre d'un scénario du diagramme des cas d'utilisation, les différents scénarios envisageables dans notre cas sont décrits dans les sections qui suivent.

a) inscription utilisateur

Ce diagramme illustre les différentes phases de l'inscription d'un utilisateur. Comme est illustré par la figure III.3, le système exige une base de données emmagasinant les données échangées. Une fois demandant une inscription, l'utilisateur aura un formulaire à remplir et choisir son rôle. Une fois rempli correctement, les données du formulaire seront stockées dans la base de données.

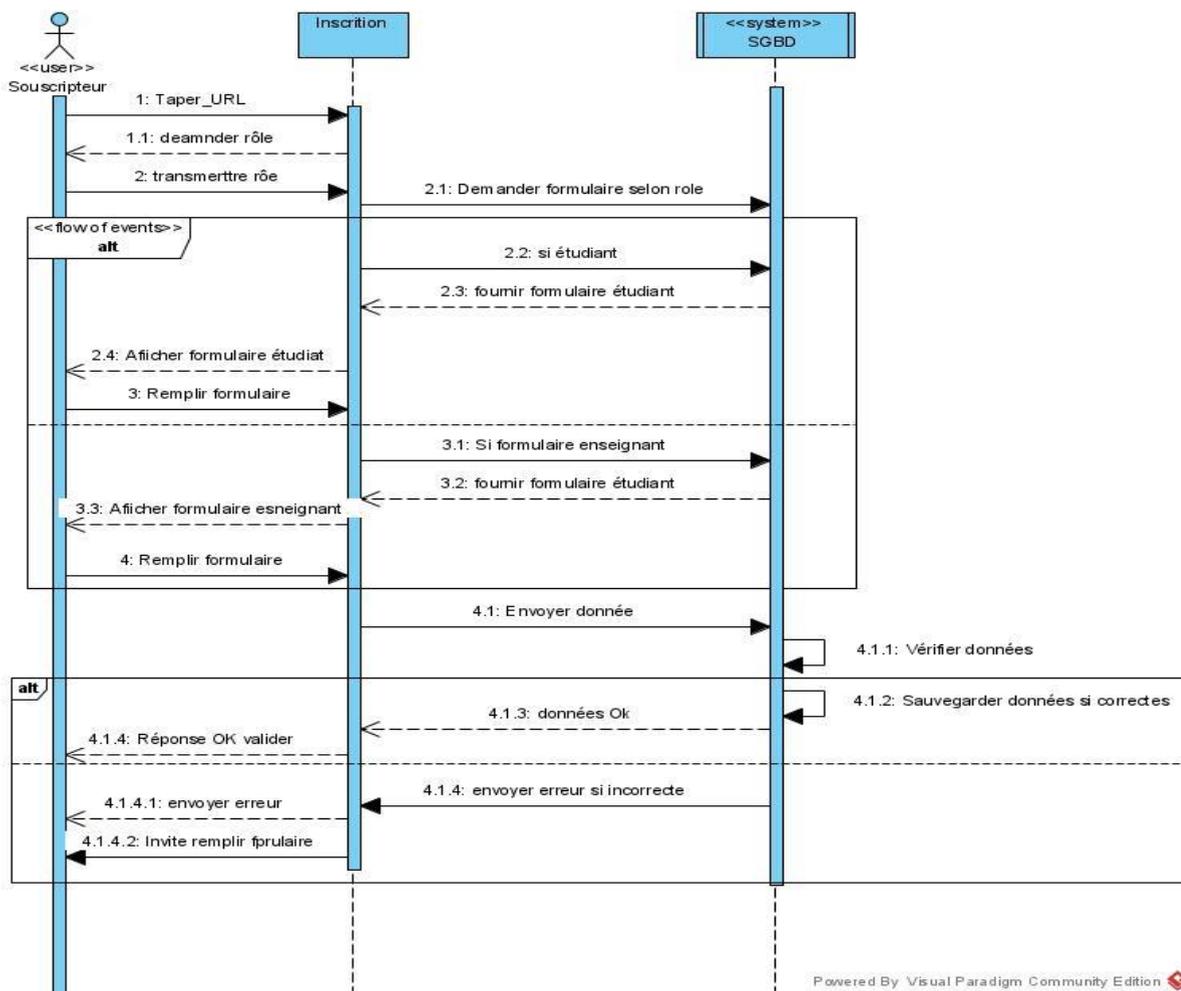


Figure III.3:Diagramme de séquence (inscription utilisateur)

b) Réalisation TP(étudiant)

L'étudiant, après s'être authentifié, aura la page du TP choisi, limité par un temps d'exécution. Si le TP est disponible, l'étudiant sera redirigé vers la page de réalisation du circuit demandé afin qu'il puisse faire le montage puis relier les équipements de mesures disponibles. Une fois les résultats obtenus, il formulera le compte-rendu et déposera le TP dans l'espace réservé.

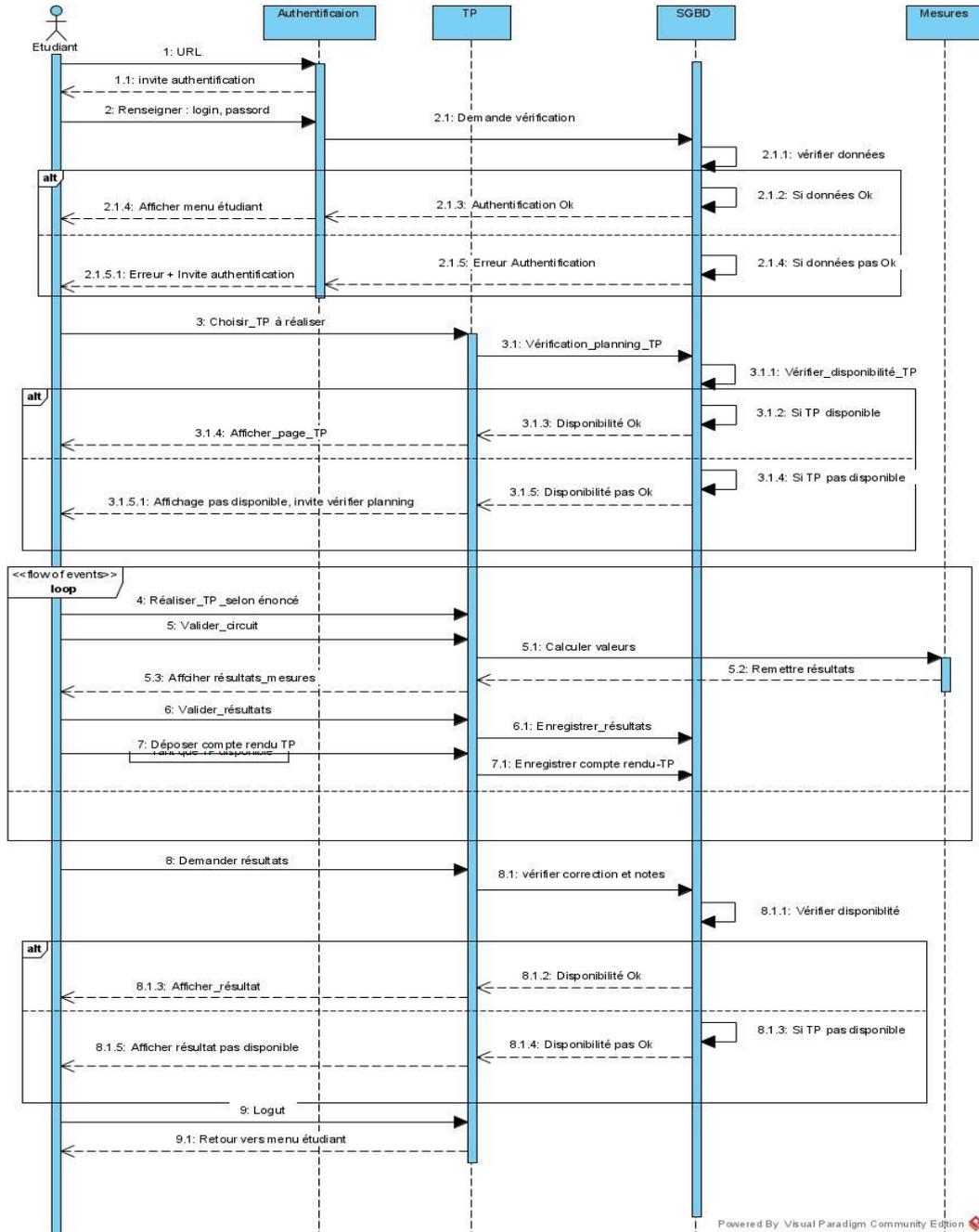


Figure III.4: Diagramme de séquence (réalisation TP étudiant)

c) Configuration du TP (enseignant)

Un TP est créé par un enseignant, qui aura à paramétrer les équipements de mesures, déposer l'énoncé et établir un planning, corriger les compte-rendu des étudiants et déposer les notes.

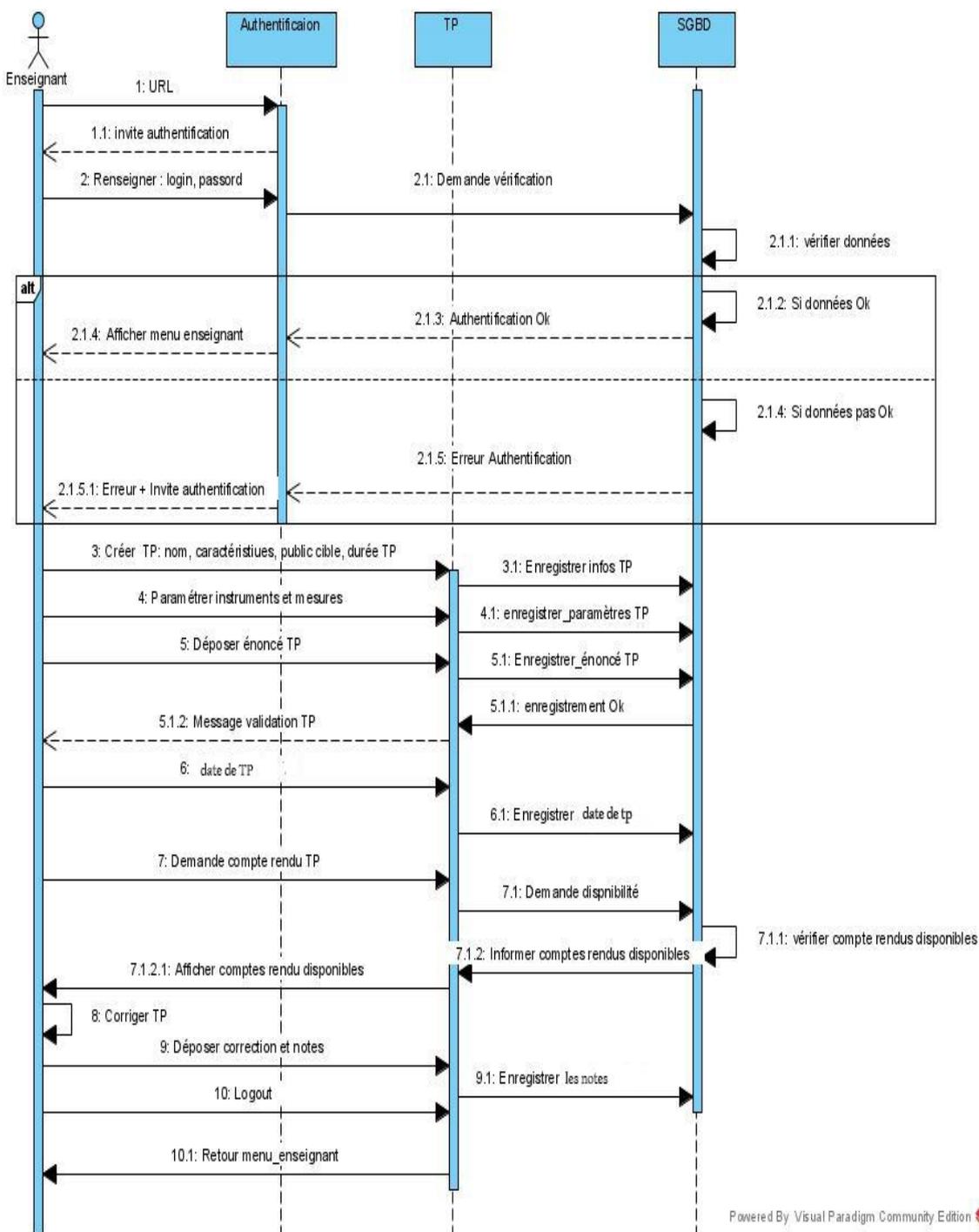


Figure III.5: Diagramme de séquence (configuration tp enseignant)

Maintenant, nous avons une idée claire sur les différentes fonctionnalités attendues ainsi que les paquetages à développer. Il est évident que le système dispose d'une partie matérielle (carte du TP, équipements de mesures) et une partie logicielle (serveur de base de données, développement des clients Web...etc.)

III.4 Réalisation du projet LABTEC

Pour se faire, l'architecture envisagée dans notre solution est illustrée par la **figure III.6** (Architecture cible) pour manipulé sur la carte TP nous utiliserons une Raspberry Pi (serveur principale) ,et un autre serveur pour les instruments de mesures .

un logiciel back-end totalement programmé avec Python en combinaison avec un logiciel front-end développé en Bootstrap , HTML, CSS ,SVG et JavaScript.

L'architecture de notre projet se divise en deux parties matérielle et logicielle

III.4.1 Architecture cible de LABTEC

La **figure III.6** illustre l'architecture cible de notre system LABTEC

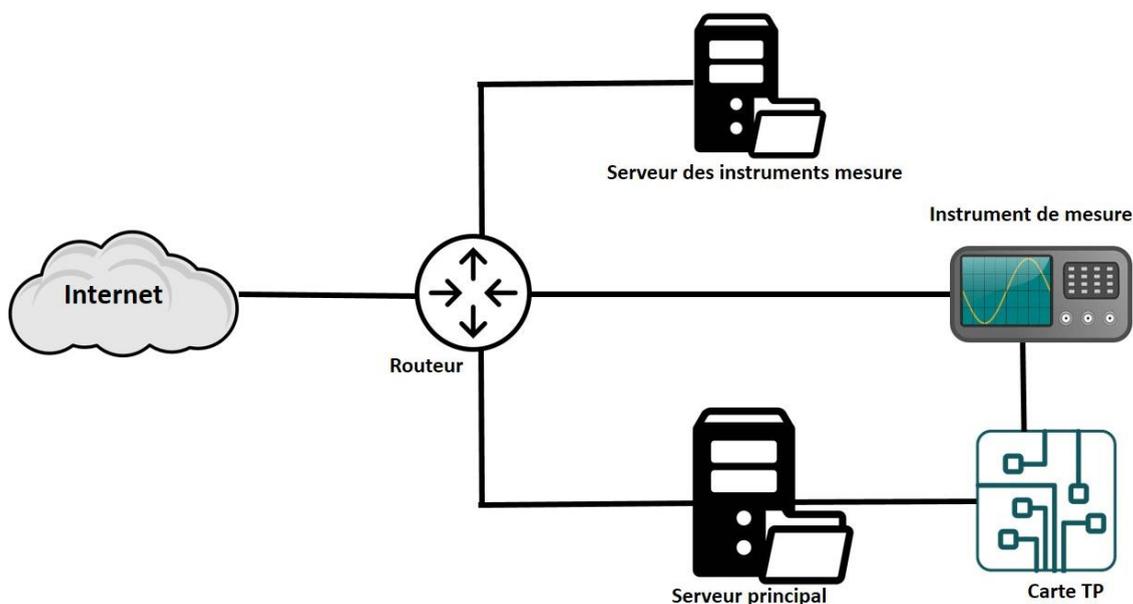


Figure III.6 : Architecture cible de LABTEC

Serveur principal

Le serveur principal du laboratoire. Il contient la base de données et toutes les opérations nécessaires pour gérer le système

Serveur des instruments mesures

Ce serveur partage l'image d'écran des instruments de mesure (pour notre système **)

Instrument de mesure

Appareils de mesure avec ou sans serveur web intégré

Carte TP

Constitué de circuit électronique et la matrice de commutation et commander via le serveur principal

III.4.2 Les outils d'implémentations

Nous avons utilisé de nombreuses technologies et outils. Sont divisés en deux types Les outils utilisés dans la Plate-forme Matérielle et les outils utilisés dans la plate-forme logicielle

III.4.2.1 Les outils de la Plate-forme Matérielle

La partie matérielle comprend :

- 01 Raspberry Pi 4
- 01 Carte TP avec une matrice de commutation
- Les instruments de mesure
- PC (serveur instrument)

a)Raspberry Pi

Le raspberry Pi servira pour héberger le serveur web qui est le point focal de notre système.

Serveur web →Raspberry→ carte relais

Le Serveur web servira aussi pour contrôler les GPIO. .

➤ Description de la carte Raspberry Pi

Le Raspberry Pi est un ordinateur dont les particularités sont la très petite taille (la taille d'une carte de crédit) et le prix très modéré (14000 DA).Il a été créé par l'anglais David Braben, dans le cadre de sa fondation Raspberry Pi, dans le but d'encourager l'apprentissage de la programmation informatique. Pour la petite histoire, raspberry signifie framboise en anglais .Le Raspberry Pi rappelle quelque peu l'Arduino, un circuit imprimé -

dont les plans sont publiés sous licence libre - sur lequel se trouve un microcontrôleur programmable et objet fétiche des partisans de l'open hardware. [28]

Tableau III.1 : Comparaison des spécifications des modèles du Raspberry Pi.

Modèle	Processeur	RAM	Les interfaces USB2.0	Ethernet	Entrées/sorties
Modèle A	ARM-v6	256 Mo	1	0	GPIO 26 pts
Modèle B	ARM-v6	512 Mo	2	1	GPIO 26 pts
Pi-2 Modèle B	ARM-v7	1Go	4	1	GPIO 40 pts

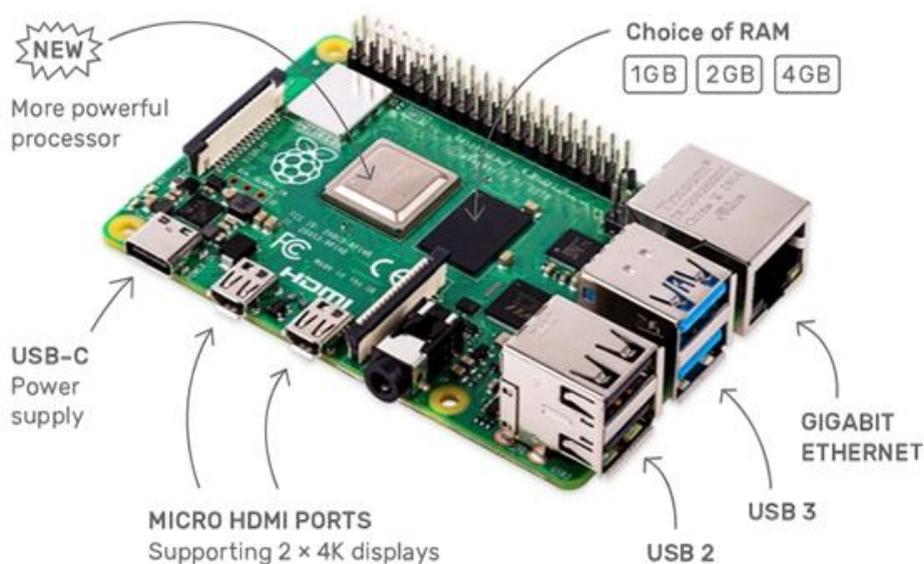


Figure III.7 : carte Raspberry Pi 4 Modèle B

b) Carte TP et Matrice de commutation

La carte de TP est conçue de manière à offrir à l'utilisateur distant les mêmes possibilités que l'utilisateur en présentiel.

Tout le circuit peut être conçu selon les règles suivantes :

- Chaque sortie de Raspberry (GPIO) est considérée comme switches
- Chaque circuit a son propre design SVG

Dans notre projet, nous avons conçu un circuit de redressement et filtrage monophasé avec un transformateur pointe milieu 12VAC, 36VA (voir **figure III.8**).

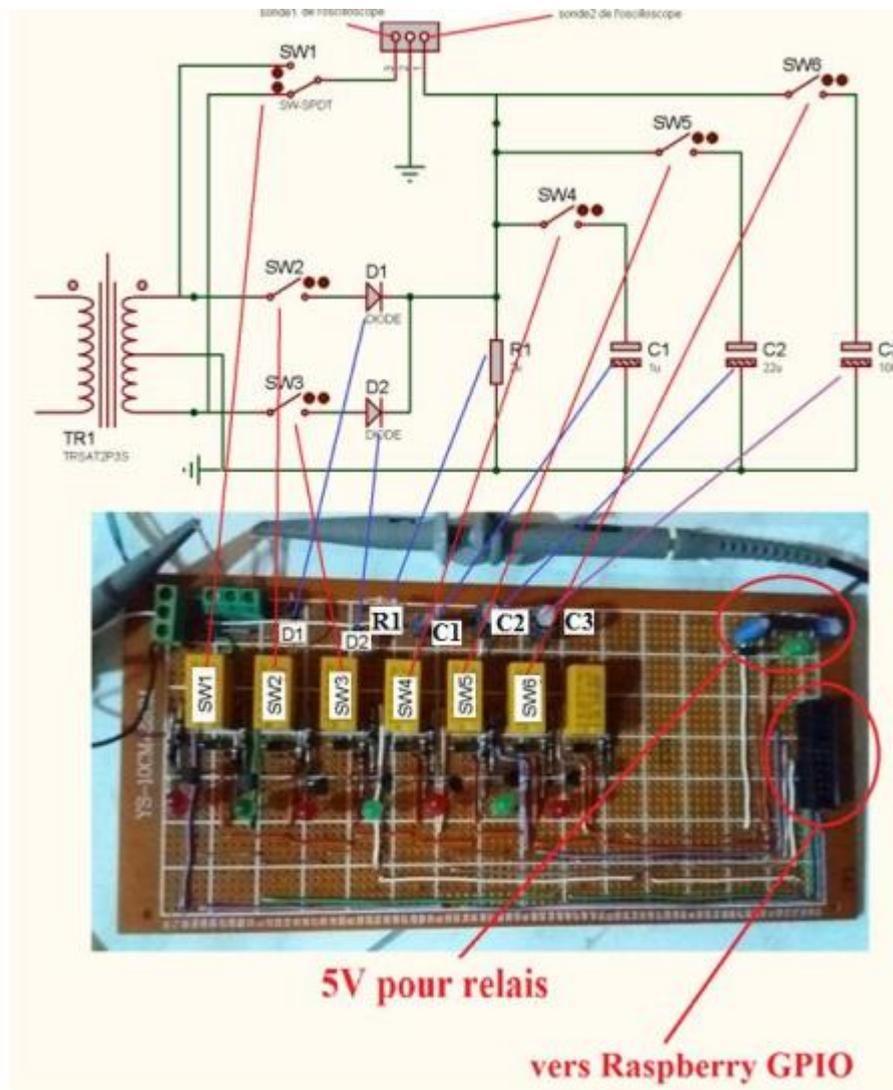


Figure III.8 : carte TP et Matrice de commutation

c) Instruments de mesure

L'instrument de mesure utilisé dans ce TP est compatible avec la norme LXI, Il est pilotable par le réseau informatique (connecteur RJ 45) mais il ne dispose pas d'un serveur embarqué.

Oscilloscope PeakTech 1240

L'oscilloscope numérique PeakTech 1240 est de dernière génération avec écran couleur rétroéclairé haute résolution, large bande et fréquence d'échantillonnage élevée, sortie VGA, mémoire interne importante, port USB et connectivité LAN pour l'intégration réseau.

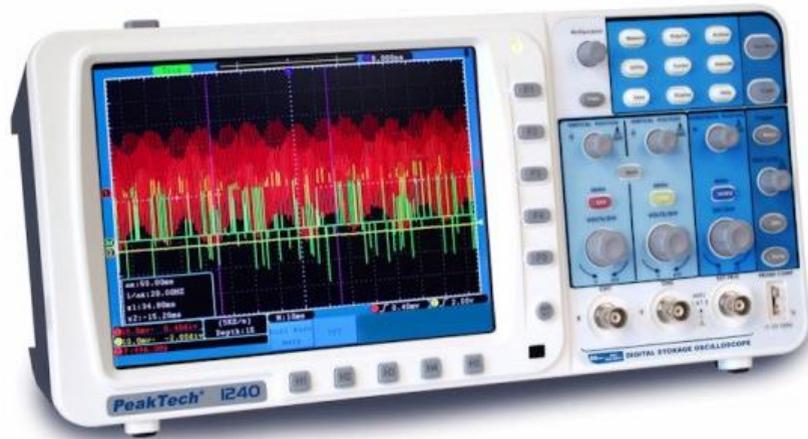


Figure III.9: Oscilloscope PeakTech 1240

Interfaces

-Hôte USB pour connecter des périphériques de stockage USB: enregistrer des captures d'écran de l'écran actuel, des formes d'onde pour un traitement ultérieur ou des paramètres utilisateur sur un périphérique de stockage USB

-Port USB pour connexion au PC: transférer les données de mesure directement sur le PC connecté et bénéficier des possibilités de stockage presque illimitées

-Sortie VGA: pour connecter l'oscilloscope PeakTech à un moniteur externe, un téléviseur ou un projecteur pour un grand affichage des formes d'onde et des mesures.

-Connexion LAN: utiliser l'oscilloscope PeakTech à distance via une connexion LAN dans le réseau de l'entreprise pour activer un large éventail d'applications.



Figure III.10 : Interfaces PeakTech 1240

d) Serveur instrument de mesure

Ce serveur doit contenir un système d'exploitation Windows et un logiciel oscilloscope qui permette de partager résultat de mesure de l'instrument

III.4.2.2 Les outils de la plate-forme logicielle

Le côté logiciel représente 80% du projet, auquel nous attachons une grande importance.

Au départ, le défi était de choisir les programmes appropriés pour développer l'application et effectuer toutes les tâches requises.

Le 26 avril 2020, on a fait une réunion avec les enseignants et leur suggestion était d'utiliser « node.js » pour développer un serveur Web, Malgré cela ils nous ont laissé une totale liberté de choisir les programmes et langages de programmation appropriés.

Après de longues recherches et consultations, nous avons choisi d'avoir le langage de programmation Python comme principal langage pour le développement du serveur Web.

a. python

Python est un langage de programmation puissant et facile à apprendre. Il dispose de structures de données de haut niveau et permet une approche simple mais efficace de la programmation orientée objet. Parce que sa syntaxe est élégante, que son typage est dynamique et qu'il est interprété, Python est un langage idéal pour l'écriture de scripts et le développement rapide d'applications dans de nombreux domaines et sur la plupart des plateformes. [32]

L'interpréteur Python et sa vaste bibliothèque standard sont disponibles librement, sous forme de sources ou de binaires, pour toutes les plateformes majeures depuis le site Internet <https://www.python.org/> et peuvent être librement redistribués. Ce même site distribue et pointe vers des modules, des programmes et des outils tiers. Enfin, il constitue une source de documentation.[32]

Python peut être utilisé pour écrire des serveurs Web très efficacement. On sait qu'il existe de nombreux frameworks et bibliothèques populaires et excellents tels que Django et Flask, qui permettent aux développeurs back-end de se concentrer sur la logique métier et de gagner beaucoup de temps sur le codage.

Pour nous, nous avons choisi d'avoir Django l'environnement avec lequel le serveur web sera développé.

b. Présentation de django

Django est un framework python open-source consacré au développement web 2.0 . Les concepteurs de Django lui ont attribué le slogan suivant: " Le framework web pour les perfectionnistes sous pression ". Il est donc clairement orienté pour les développeurs ayant comme besoin de produire un projet solide rapidement et sans surprise ... c'est à dire à tous les développeurs !

Comme il est toujours compliqué de ne partir de rien, Django propose une base de projet solide. Django est donc une belle boîte à outils qui aide et oriente les développeurs dans la construction de leurs projets.

Pour la petite histoire Django a vu le jour en 2003 et a été publié sous licence BSD en juillet 2005.[35]

➤ **Pourquoi choisi le framework Django?**

L'une des principales raisons du choix de cet environnement est notre expérience avec Python. Beaucoup de développeurs sont beaucoup plus productifs sur python que sur n'importe quel autre langage. La technologie derrière ce framework peut être une bonne raison, mais Django est une raison à part entière. Tout est pensé de base. Inutile de réinventer la roue à chaque projet web: les bases sont déjà présentes. Il nous fournira les outils nécessaires à sécuriser l'application, à gérer la structure de des modèles avec un ORM (Object-relation mapping) , etc.

➤ **La structure Django**

Django s'inspire du modèle MVC (disons plutôt MVT), c'est-à-dire que la structure du framework sépare les données (models) qui sont séparées des traitements (controller) qui sont eux-mêmes séparés de la vue (view / template). On nous oblige à bien coder, une structure doit être respectée et cela ne peut être que profitable au travail collaboratif ou simplement la cohérence / communication entre différents projets.

Le moteur de template de base est simple, efficace souple et facile à prendre en main. Un routeur permet de rediriger les actions en fonctions des URL et une API permet de fournir des informations sur le projet sans passer par la case SQL. D'ailleurs l'ORMinclu éloignera de toute requête SQL.

Chaque projet Django propose de lancer son propre serveur web et d'y faire à peu près tout ce que l'on veut dans un environnement de test.

Un des concepts des plus intelligent de Django est de proposer un espace admin tout fait dans ses contrib . Une fois les modèles créés (la structure du projet), on accède à une interface web CRUD en quelques minutes.

on peut personnaliser cet espace admin ou créer des propres templates et des propres formulaires . Django propose des outils pour sécuriser les données ou afficher les erreurs si besoin.



Figure III.11 :Django MVT: Architecture d'un projet

c. Framework flask

Flask est un framework d'application Web WSGI léger. Il est conçu pour rendre la mise en route rapide et facile, avec la possibilité de s'adapter à des applications complexes. Il a commencé comme un simple wrapper autour de Werkzeug et Jinja et est devenu l'un des frameworks d'application Web Python les plus populaires.

Flask propose des suggestions, mais n'applique aucune dépendance ni disposition de projet. Il appartient au développeur de choisir les outils et les bibliothèques qu'il souhaite utiliser. Il existe de nombreuses extensions fournies par la communauté qui facilitent l'ajout de nouvelles fonctionnalités.

On a utilisé cette framework pour créer le serveur des instruments mesures

d. AJAX (Asynchronous Javascript and XML)

AJAX est l'acronyme d'Asynchronous JavaScript and XML, ce qui, transcrit en français, signifie « JavaScript et XML asynchrones ».

Derrière ce nom se cache un ensemble de technologies destinées à réaliser de rapides mises à jour du contenu d'une page Web, sans qu'elles nécessitent le moindre rechargement visible par l'utilisateur de la page Web. Les technologies employées sont diverses et dépendent du type de requêtes que l'on souhaite utiliser, mais d'une manière générale le JavaScript est constamment présent.

D'autres langages sont bien entendu pris en compte comme le HTML et le CSS, qui servent à l'affichage, mais ceux-ci ne sont pas inclus dans le processus de communication. Le transfert de données est géré exclusivement par le JavaScript, et utilise certaines technologies de formatage de données, comme le XML ou le JSON

e. Technologies utilisées dans le front-end

➤ **HTML** (HyperText Markup Language)

L'HTML est un langage informatique utilisé sur l'internet. Ce langage est utilisé pour créer des pages web. L'acronyme signifie HyperText Markup Language, ce qui signifie en français "langage de balisage d'hypertexte". Cette signification porte bien son nom puisqu'effectivement ce langage permet de réaliser de l'hypertexte à base d'une structure de balisage.

Ce n'est pas à proprement parlé un langage de programmation, mais plutôt un langage qui permet de mettre en forme du contenu. Les balises permettent de mettre en forme le texte et de placer des éléments interactifs, tel des liens, des images ou bien encore des animations. Ces éléments ne sont pas dans le code source d'une page codé en HTML mais "à côté" et la page en HTML ne fait que reprendre ces éléments.

Pour visualiser une page en HTML il est nécessaire d'utiliser un navigateur web.

➤ **SVG**

Le circuit électronique de l'interface web est réalisé par un nouvel élément SVG servant au dessin de graphiques. Il s'agit d'un espace de pixels initialement transparents, armés de JavaScript pour réaliser un bon nombre de fonctions graphiques.

Ses avantages est qu'il fonctionne très bien avec les autres standards (SVG,JavaScript) et il est performant et accéléré matériellement sur la plupart des navigateurs et systèmes

➤ **JavaScript**

JavaScript est un langage de programmation de scripts principalement utilisé dans les pages web interactives. C'est un langage orienté objets à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas

des instances de classes, mais qui sont chacun équipés de constructeurs permettant de générer leurs propriétés, et notamment une propriété de prototypage qui permet d'en générer des objets héritiers personnalisés [11].

➤ **Bootstrap**

Bootstrap est un framework CSS permettant de faciliter l'écriture de la mise en forme CSS d'un site web.

Plus généralement, nous pouvons dire que c'est un rassemblement de différents morceaux de code utiles à la création d'une page web, aussi bien dans la mise en forme CSS que dans l'animation et l'interactivité du contenu JavaScript.

f. Système de gestion de base de données (SGBD)

SQLite est un système de base de données ou une bibliothèque proposant un moteur de base de données relationnelles. Il repose sur une écriture en C, et sur une accessibilité via le langage SQL (Structured Query Language). SQLite présente la particularité d'être directement intégré aux programmes et dans l'application utilisant sa bibliothèque logicielle alors que ses concurrents comme MySQL reproduisent de leur côté le schéma classique client-serveur. Avec SQLite, la base de données est intégralement stockée dans un fichier indépendant du logiciel.

Créé au début des années 2000 par D. Richard Hipp, SQLite propose un accès plus rapide aux données, mais aussi plus structuré et avec davantage de sécurité.

III.4.3 Réalisations et discussions.

III.4.3.1 Première solution proposée, utiliser Django et node.js

Notre objectif est maintenant de construire un serveur Web permettant à l'utilisateur de contrôler et commander les circuits électronique. Où nous concevons ces circuits du côté front-end en utilisant SVG. L'utilisateur interagit avec le circuit dans le navigateur en utilisant le langage de programmation JavaScript.

Ensuite, des instructions sont envoyées au serveur Web pour effectuer les modifications nécessaires

,le problème maintenant est donc de savoir comment envoyer des données à partir d'une page Web vers le serveur 'raspberry' et commander les sorties GPIO

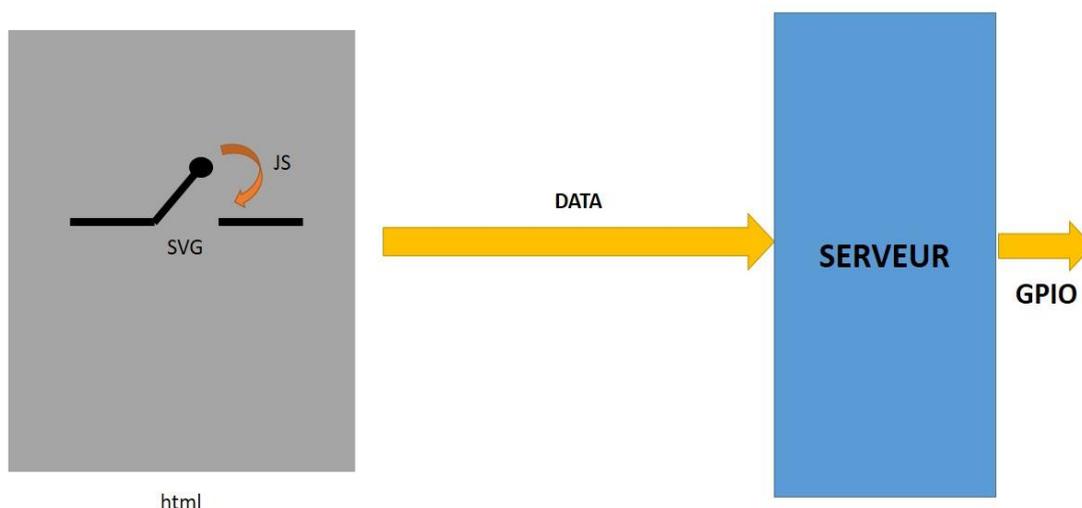


Figure III.12 : Flux de données du système proposé

Nous cherchons toujours le moyen le plus simple. puisque l'utilisateur réalise le circuit électronique via le navigateur en utilisant 'JavaScript'. Nous savons que la bibliothèque 'node.js' est construite en langage 'JavaScript' et permet de commander facilement les GPIOs qui sont situés dans la carte raspberry, donc la solution que nous avons proposée au début était la suivante:

Nous avons installé 'node.js' sur le serveur, puis nous avons créé un fichier 'Pi_setting.js', ce fichier contient des fonctions js (Les fonctions node.js), chaque fonction représente les configurations d'un circuit électronique (Les états possibles des relais), ce fichier est importé au l'html. Ensuite, nous appelons la fonction appropriée puis nous saisissons les valeurs requises

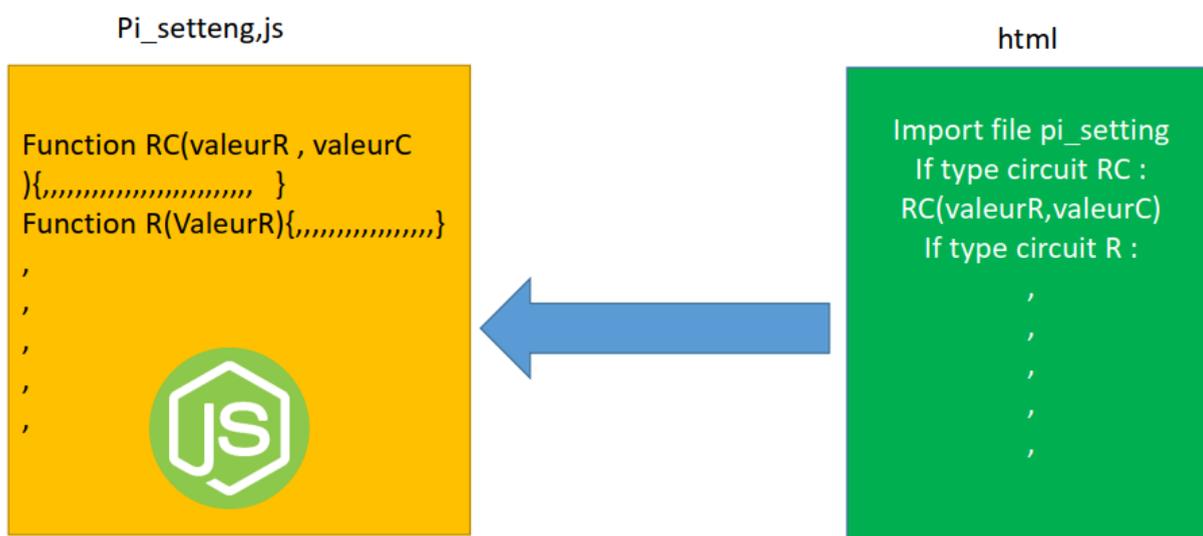
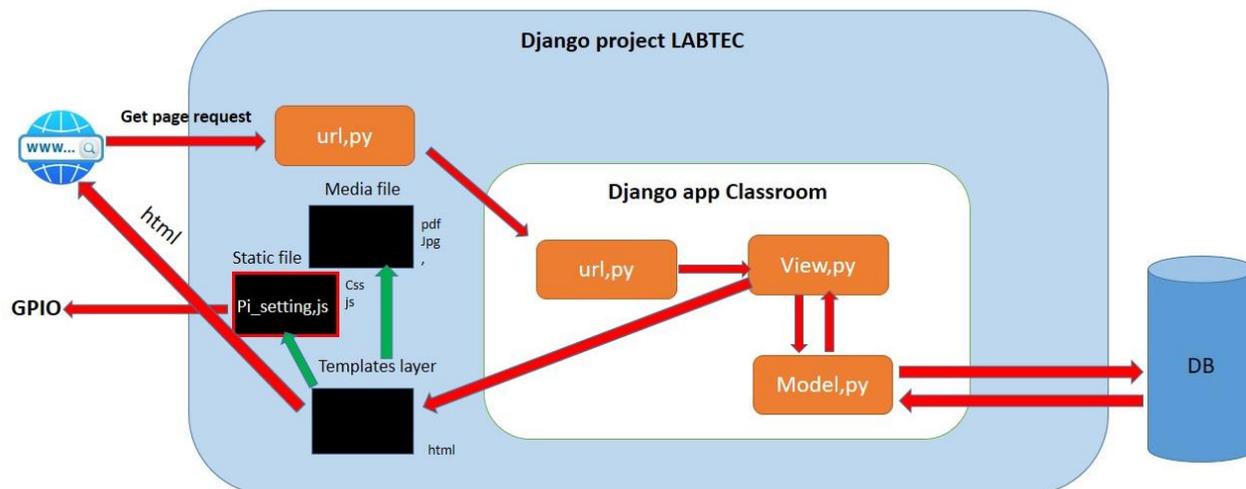


Figure III.13 : Structure du fichier Pi_setting.js

A partir de cela, nous pouvons dessiner l'architecture globale de notre système (Figure III.14)

**Figure III.14** : Architecture logicielle de la solution (première variante)

Après avoir essayé cette solution, nous avons découvert qu'il n'est pas possible d'utiliser les codes d'une bibliothèque 'node.js' sans faire fonctionner son serveur 'node.js', donc la seule solution pour travailler avec cette méthode est d'utiliser un serveur Web 'node.js' à côté d'un serveur Web 'Django' cela sera difficile et causera de nombreux problèmes de sécurité.

Pour plus de détails sur ce problème, on peut consulter le lien [41]

Après l'échec de la première solution, nous devons penser à une autre manière en utilisant le langage de programmation Python, c'est-à-dire sans utiliser de nombreux programmes qui peuvent causer des problèmes dans le système

III.4.3.2 Deuxième solution « Utilisation des bibliothèques Python »

Dans Python, il existe de nombreuses bibliothèques qui aident à contrôler les gpio de Raspberry.

Parmi toutes ces bibliothèques RPi.GPIO sont les plus faciles et les plus utilisées. ce package fournit une classe pour contrôler le GPIO sur un Raspberry Pi.

Exemple

Pour contrôler une LED connectée à GPIO18, on utilise ce code:

```
1 import RPi.GPIO as GPIO
2 import time
3 GPIO.setmode(GPIO.BCM)
4 GPIO.setwarnings(False)
5 GPIO.setup(18,GPIO.OUT)
6 print "LED on"
7 GPIO.output(18,GPIO.HIGH)
```

Nous allons créer une fonction dans un fichier `views.py`. Cette fonction effectue les réglages nécessaires pour le circuit électronique. Par exemple, si l'utilisateur ferme le switch numéro 1, la fonction ouvre le GPIO. Mais le problème maintenant est de savoir comment envoyer des données d'une Gabarits 'Template' vers `views.py`

Pour ce faire, il existe plusieurs solutions, mais la meilleure solution pour nous c'est utiliser une URL au début les données (Data) qui varie en utilisant 'JavaScript' et 'SVG' puis nous orientons l'utilisateur vers le lien où il trouve ces valeurs ; par exemple `www.exemple.com/?data=100000`.

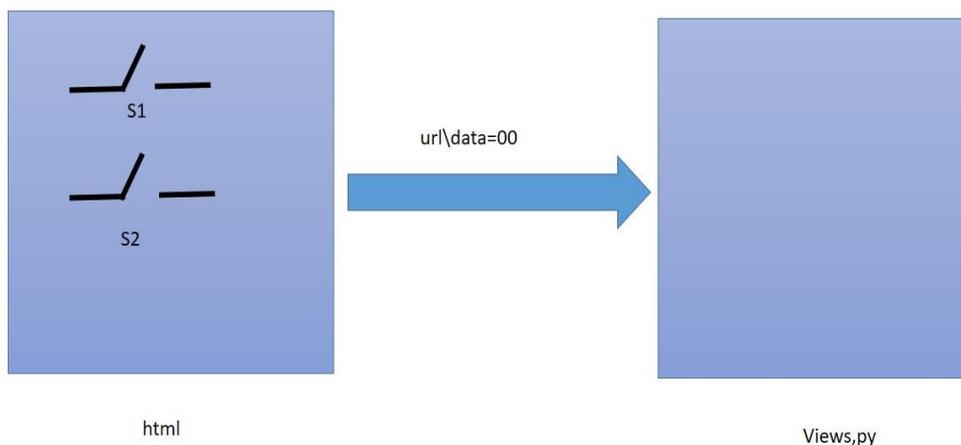


Figure III.15 : lien entre l'HTML et Views.Py

Dans un fichier `Views.py` nous créons une fonction qui manipule ces valeurs. Avec l'aide d'une bibliothèque `RPI.GPIO`. Par exemple, la fonction sera comme ceci

```
▼ def setting_gpio (request , data):  
    import RPI.gpio as GPIO  
    // data = 00  
    // Changer le statut du GPIO  
    GPIO.setmode(GPIO.BCM) #  
    GPIO.setup(17, GPIO.OUT) #  
    GPIO.setup(18, GPIO.OUT) #  
▼ if data[0] == 1 :  
    |     GPIO.output(17, GPIO.HIGH)  
▼ else :  
    // Ajouter de nombreux paramètres  
    .  
    .  
    .  
    return render(request , 'page.html')
```

Remarque : La fonction est liée à url.py. Et de nombreux réglages sont effectués, nous voulons juste expliquer ce qui est important.

Après avoir essayé cette solution, nous sommes tombés dans un autre problème, qui est les « chemins infaisables ». En effet, si l'utilisateur ouvre S1 il est entré dans une boucle sans fin et donc la page suivante n'apparaît pas . Parce que GPIO.output(17, GPIO.HIGH) est une fonction qui contient une boucle .

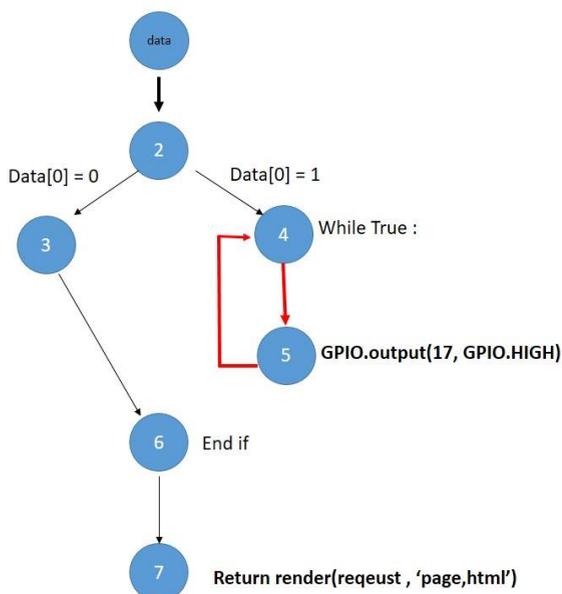


Figure III.16 : Le graph CFG de la fonction setting_GPIO

Le chemin infaisable de ce graphe est « 1-2-4-5 ».

D'où la nécessité de chercher une autre solution afin de contourner ce problème.

III.4.3.3 Troisième solution : utilisation d'un OS (Operating System)

Lors de notre recherche d'une solution au problème. Nous avons trouvé un moyen de contrôler le GPIO via Terminal. Autrement dit, via le système d'exploitation. C'est en utilisant bibliothèque WiringPi

Bibliothèque « WiringPi »

« WiringPi » est l'une des bibliothèques incontournables pour tous ceux qui utilisent le Raspberry Pi avec des cartes d'extension.

Développée par Gordon, cette bibliothèque est écrite en C. Elle contient des routines permettant un accès facile à une partie des périphériques les plus populaires. [9,11]

L'utilitaire GPIO

WiringPi est livré avec un programme séparé pour aider à gérer l'interface GPIO intégrée ainsi que des modules supplémentaires tels que le PiFace et d'autres périphériques comme le Gertboard ainsi que des périphériques génériques de type Expander GPIO.

Ce programme, appelé `gpio`, peut également être utilisé dans des scripts pour manipuler les broches GPIO - définir les sorties et lire les entrées. Il est même possible d'écrire des programmes entiers en utilisant simplement la commande `gpio` dans un 'shell-script', même si ce n'est pas très efficace de le faire de cette façon. Une autre façon de l'appeler est d'utiliser la 'fonction `system ()` en C / C ++' ou c'est équivalent dans d'autres langages de programmation.

La commande `gpio` est conçue pour être installée en tant que programme `setuid` et appelée par un utilisateur normal sans utiliser la commande «`sudo`» ou se connecter en tant que `root`.

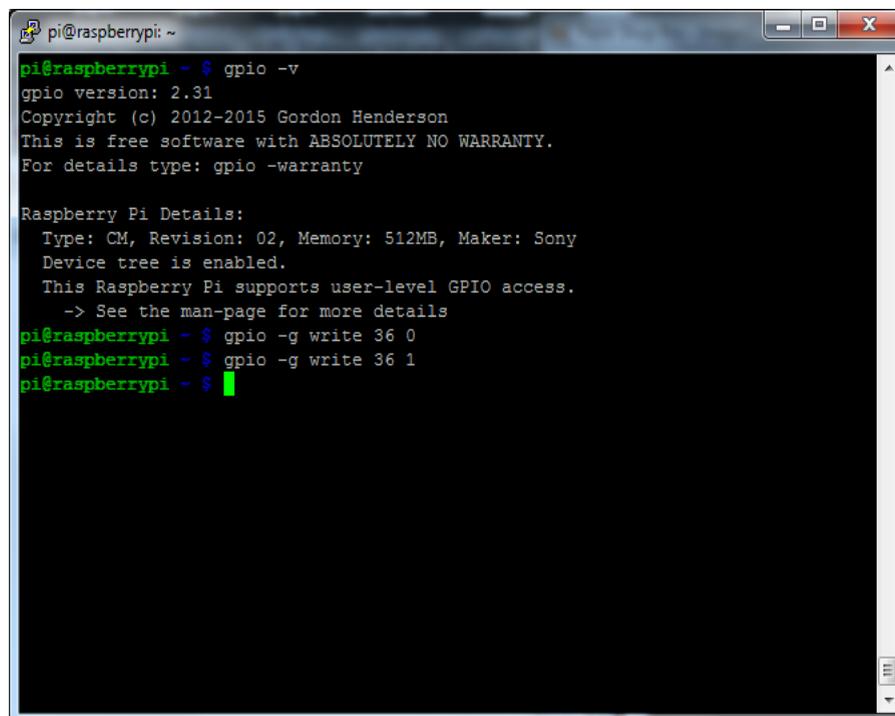
Commandes d'entrée et de sortie standard

`gpio mode <pin> in/out/pwm/clock/up/down/tri`

Cela définit le mode d'une broche en mode entrée, sortie, pwm ou horloge, et peut en outre régler les résistances de tirage haut / bas internes sur pull-up, pull-down ou aucun.

`Gpiowrite<pin> 0/1`

Ceci définit une broche de sortie sur haut (1) ou bas (0)



```
pi@raspberrypi: ~  
pi@raspberrypi ~ $ gpio -v  
gpio version: 2.31  
Copyright (c) 2012-2015 Gordon Henderson  
This is free software with ABSOLUTELY NO WARRANTY.  
For details type: gpio -warranty  
  
Raspberry Pi Details:  
Type: CM, Revision: 02, Memory: 512MB, Maker: Sony  
Device tree is enabled.  
This Raspberry Pi supports user-level GPIO access.  
-> See the man-page for more details  
pi@raspberrypi ~ $ gpio -g write 36 0  
pi@raspberrypi ~ $ gpio -g write 36 1  
pi@raspberrypi ~ $
```

Figure III.17: Commandes d'entrée et de sortie standard (`gpio`)

III.4.3.4 python et le système d'exploitation (Operating System)

Python nous fournit des bibliothèques pour accéder aux ressources système. Tel que subprocess (Gestion de sous-processus) .

Le module subprocess permet de lancer de nouveaux processus, les connecter à des tubes d'entrée/sortie/erreur, et d'obtenir leurs codes de retour. Ce module a l'intention de remplacer plusieurs anciens modules et fonctions [33] :

```
os.system*  
os.spawn*  
os.popen*  
popen2.*  
commands.*
```

Utilisation du module subprocess

La méthode recommandée pour lancer des sous-processus consiste à utiliser les fonctions pratiques suivantes.

```
subprocess.call(args, *, stdin=None, stdout=None, stderr=None, shell=False)
```

exemple :

```
>>> subprocess.call(["ls", "-l"])  
0  
  
>>> subprocess.call("exit 1", shell=True)  
1
```

Figure III.18 : exemple d'utilisation du module « subprocess »

Nous utilisons cette bibliothèque dans un fichier views et la structure des données serait comme suit (**figure III.19**)

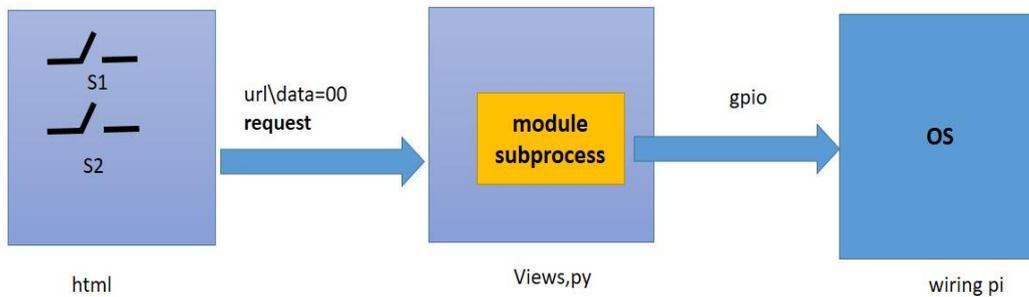


Figure III.19 : le module subprocess dans notre système

Voici à quoi ressemblera un script en views.py pour 6 switches

```

1
2 @login_required
3 @student_required
4- def get_vr(request, id , data):
5     session_key = request.session.session_key
6-     if Session.objects.all().count() > 1 :
7-         if str(session_key) == str(Session.objects.all()[0]) :
8
9             cartTP('000000') # initialisation
10            request.session.set_expiry(50)
11
12            cartTP(data)
13            return render(request, 'classroom/students/loadig.html')
14-         else :
15             return render(request, '500.html')
16-     else :
17         cartTP(data)
18         return render(request, 'classroom/students/loadig.html')
19
20
21- def cartTP(statuspin) :
22     #print('statuspin ===== ', statuspin)
23     switch1 = [statuspin[0] , '17' ]
24     switch2 = [statuspin[1] , '27' ]
25     switch3 = [statuspin[1] , '22' ]
26     switch4 = [statuspin[1] , '10' ]
27     switch5 = [statuspin[1] , '09' ]
28     switch6 = [statuspin[1] , '11' ]
29     switchall = [switch1 , switch2, switch3 , switch4 , switch5 , switch6]
30     #switch7 = statuspin[6]
31-     for switch in switchall :
32         subprocess.call(['gpio' , '-g' , 'mode' , switch[1] , 'out'])
33         subprocess.call(['gpio' , '-g' , 'write' , switch[1] ,switch[0]])
34
35

```

Figure III.20: Développement de la fonction « carte TP »

III.4.3.5 Améliorations ajoutées au système

un autre problème dans notre système est que nous ne pouvons pas envoyer des 'instructions au serveur sauf en allant sur la page de confirmation à chaque fois, si l'utilisateur a besoin de changer la forme du circuit électronique, il doit appuyer sur le bouton de confirmation

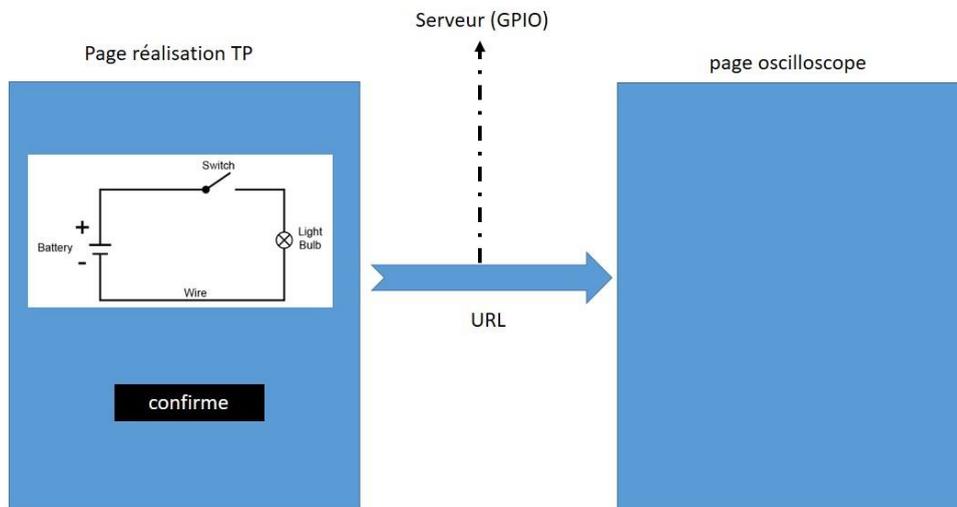


Figure III.21: Changement de l'état du GPIO avec confirmation

Pour améliorer notre système nous voulons envoyer des commandes directement depuis la page de réalisation. lorsque l'utilisateur ferme et ouvre les switch. Les commandes sont envoyées en temps réel (command GPIO) sur la même page.

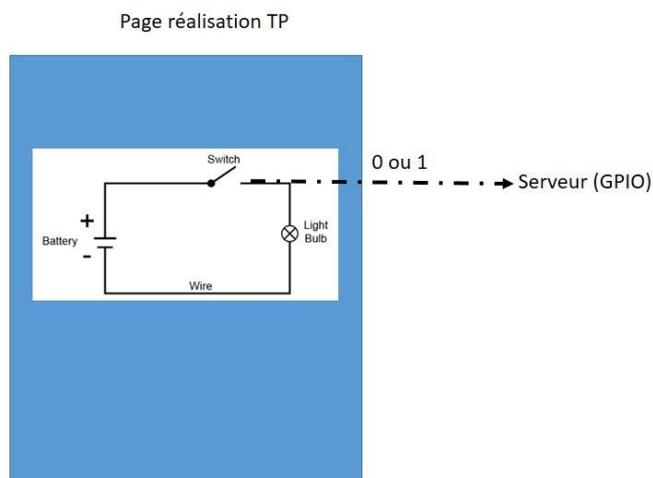


Figure III.22: Commande GPIO sur la même page

L'utilisation de la technologie AJAX permet de commander directement GPIO en temps réelle sur la même page.

Comment AJAX fonctionne dans Django ?

AJAX n'est rien d'autre qu'une combinaison d'objets JavaScript et XHR(XMLHttpRequest). Le concept est simple:

A - Code JavaScript sur le client - Côté / navigateur fait une demande lorsqu'un événement se produit sur la page Web. Le code JavaScript générera un objet XHR et est envoyé en tant qu'objet de requête au serveur. L'objet XHR contient des données / objets JavaScript. L'objet XHR contient également l'URL ou le nom de la fonction de rappel sur le serveur.

Que notre exemple nous envoie les données suivantes le nom switch (switch_1). Et le cas est soit 0 soit 1 données sous forme json

```

405 <script>
406 function myfnc(switch_num,statu_switch){
407     $.ajax({
408         type: "POST",
409         url: "/students/like",
410         data: {'switch_num': switch_num, 'statu_switch': statu_switch},
411         dataType: "json",
412
413     });
414 }
415 </script>

```

Utiliser url « /student/like » qui est associé à la fonction qui commander la GPIO line 17 en url.py

```
6  urlpatterns = [  
7      path('', classroom.home, name='home'),  
8  
9      path('students/', include([  
10         path('', students.QuizListView.as_view(), name='quiz_list'),  
11         path('interests/', students.StudentInterestsView.as_view(), name='student_interests'),  
12         path('taken/', students.TakenQuizListView.as_view(), name='taken_quiz_list'),  
13         path('quiz/<int:pk>/', students.take_quiz, name='take_quiz'),  
14         path('quiz/Compt_rendu/<int:id>', students.pdf_view, name='pdf_view'),  
15         path('test', students.test_redirect, name='test_view'),  
16         path('quiz/Compt_rendu/<int:id>/data=<str:r>', students.get_vr, name='get_vr'),  
17         path('like', students.like, name='like_h' ),  
18  
19         ], 'classroom'), namespace='students')),  
20
```

B - La demande est gérée par le serveur avec une fonction de rappel

La fonction d'affichage ou la fonction de rappel appropriée gère la demande. Elle enverra une réponse de succès ou d'échec. La requête étant asynchrone, le reste du code s'exécute sans interruption. à ce moment, le serveur traite la demande.

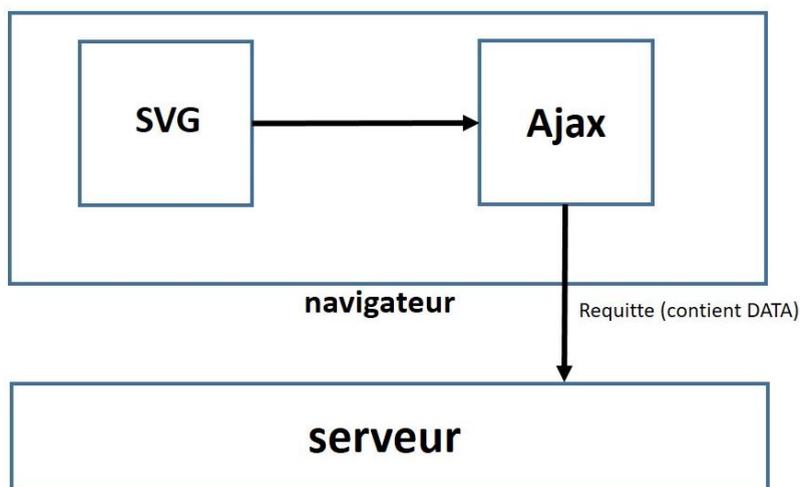


Figure III.23: Relation entre Ajax et le serveur

La fonction appelée sur le serveur est décrite ci-dessous

```
318 @login_required
319 @require_POST
320 @csrf_exempt
321 def like(request):
322     if request.method == 'POST':
323         switches = {'switch_1' : '17' , 'switch_2' : '27', 'switch_3' : '22',
324                   'switch_4' : '10' , 'switch_5' : '09' , 'switch_6' : '11' ,
325                   'switch_7' : '0000' }
326         data = request.body
327         switch = str(data).split('&')[0].split('=')[1]
328         status = str(data).split('&')[1].split('=')[1][:-1]
329         if switch == 'switch_1' and status == 1 :
330             subprocess.call(['gpio', '-g', 'mode', '5', 'out'])
331             subprocess.call(['gpio', '-g', 'write', '5', '0'])
332         if switch == 'switch_1' and status == 0:
333             subprocess.call(['gpio', '-g', 'mode', '5', 'out'])
334             subprocess.call(['gpio', '-g', 'write', '5', '1'])
335
336         subprocess.call(['gpio' , '-g' , 'mode' , switches[switch] , 'out'])
337         #print(['gpio', '-g', 'mode', switches[switch], 'out'])
338         subprocess.call(['gpio' , '-g' , 'write' , switches[switch] , status])
339         #print(['gpio', '-g', 'write', switches[switch] , status])
340
341
342         ctx = {'likes_count': 'rt', 'message': 'ffffff'}
343         # use mimetype instead of content_type if django < 5
344         return HttpResponse(json.dumps(ctx), content_type='application/json')
```

III.4.3.6 Création d'un serveur web pour l'instrument de mesure (Oscilloscope PeakTech 1240)

L'Oscilloscope PeakTech du laboratoire ne contient pas de serveur web. Il fallait donc trouver une solution

Il est important de noter l'oscilloscope en question ne fonctionne sur l'ordinateur qu'à l'aide d'un programme spécial du fabricant et il n'est pas open source. Il ne fonctionne que sur le système d'exploitation Windows. Ce programme permet uniquement de visualiser l'écran de l'oscilloscope. (Alors maintenant, nous faisons le même travail de partage de l'écran via le serveur Web principal)

La meilleure solution à laquelle nous avons pu accéder est d'utiliser un serveur Web fonctionnant sur un système d'exploitation Windows. La seule tâche de ce serveur est de partager l'écran.

Nous avons développé ce serveur en utilisant flask et une bibliothèque pyscreenshot

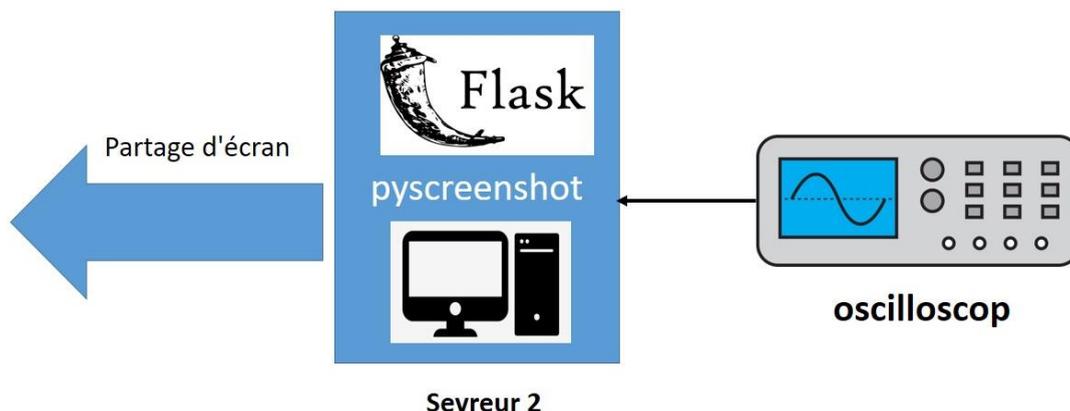


Figure III.24: serveur Flask pour accès à l'oscilloscope PeakTEch

Les fichiers et programmes du serveur sont disponibles sur github sur le lien suivant[42]

III.4.3.7 Authentification et autorisation

La gestion de plusieurs types d'utilisateurs ou rôles dans une application Web se résume à la gestion des autorisations. Les indicateurs booléens comme 'is_student' ou 'is_teacher' fonctionnent généralement comme des règles d'autorisation fixes et prédéfinies, qui sont incorporées dans le code source. il est important de souligner ces deux concepts:

- **Authentification (Connexion)**
- **Autorisation (Permission)**

L'authentification est le processus consistant à vérifier si la personne est bien celle qu'elle prétend être. Concrètement, il s'agit du processus de vérification du nom d'utilisateur et du mot de passe (connexion).

L'autorisation est le processus de vérification de ce que cette personne particulière est autorisée à faire dans l'application. Cela peut être réalisé en utilisant des décorateurs (dans view.py), le cadre d'autorisations intégré de Django ou une application tierce comme **django-guardian** ou **django-rules**.

Si vous regardez le code source [41]. Vous constaterez que nous utilisons le décorateur. Comme chaque fonction ou classe de la (view) nécessite des autorisations. Ces décorations sont créées dans un fichier decorators.py.

```

358 @login_required
359 @student_required
360 def get_vr(request, id , r):
361     print(request.session.keys())
362     session_key = request.session.session_key

```

Par exemple, pour exécuter cette fonction, l'utilisateur doit être connecté (@login_required) et être un étudiant (@student_required)

```

5 def student_required(function=None, redirect_field_name=REDIRECT_FIELD_NAME, login_url='login'):
6
7
8
9
10 actual_decorator = user_passes_test(
11     lambda u: u.is_active and u.is_student,
12     login_url=login_url,
13     redirect_field_name=redirect_field_name
14 )
15 if function:
16     return actual_decorator(function)
17 return actual_decorator
18

```

De cette façon, les décorateurs sont créés .décorateur qui vérifie que l'utilisateur connecté est un étudiant, redirige vers la page de connexion si nécessaire.

Autorisations par utilisateur

Comme on a vu, notre application contient trois types d'utilisateurs. Ce sont les autorisations pour chaque utilisateur

Autorisations d'étudiant :

- Il peut voir des nouveaux TPs (Selon les filières)
- Accès au laboratoire réel (pilotage du circuit et mesures)
- il peut voir la fiche de correction TPs et le compte-rendu
- Envoyez le fichier de TPs (compte-rendu)
- Voir les annonces pour chaque TP.

Autorisations l'enseignant

- Ajouter, modifier et supprimer le TPs
- Voir la liste des étudiants qui ont terminé leurs TP
- Évaluation des étudiants et envoi du fichier de correction(Peut être modifié)
- Afficher le calendrier du TPs
- Accès la page de réalisation TPs
- Active la notification par e-mail
- Active le calendrier de TPs
- Choisir la durée de TP et la période de l'accès au labo distant

Autorisations d'administrateur

- Ajouter et supprimer des utilisateurs
- Modifier les utilisateurs existants
- Réinitialiser les mots de passe des utilisateurs
- Attribuer le statut d'employé et / ou de super utilisateur à un utilisateur
- Ajouter ou supprimer des autorisations utilisateur
- Créer des groupes d'utilisateurs; et ajouter des utilisateurs à un group

III.4.3.8 Modèles et bases de données

Un modèle est la source de données unique et définitive à propos de vos données. Il contient les champs et le comportement essentiels des données que vous stockez. Généralement, chaque modèle correspond à une seule table de base de données.

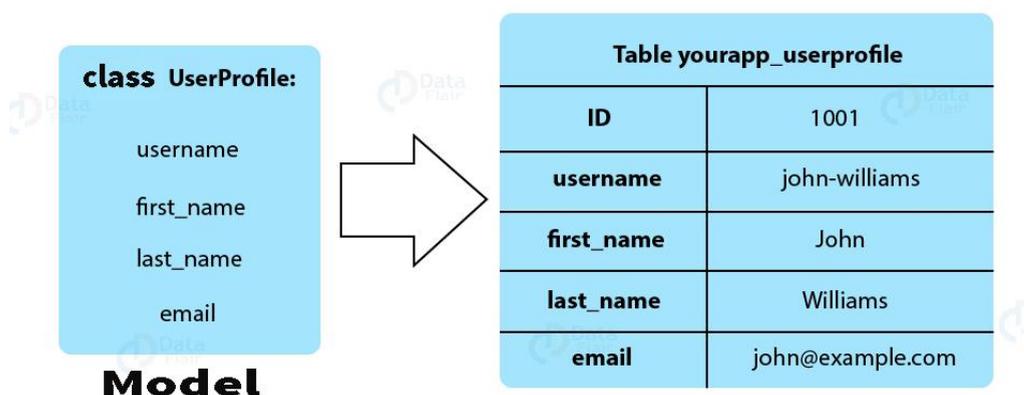


Figure III.25 : Modèles et bases de données

a) Tables de base de données et toutes les relations

Ci-dessous, la structure de la base de données de notre système

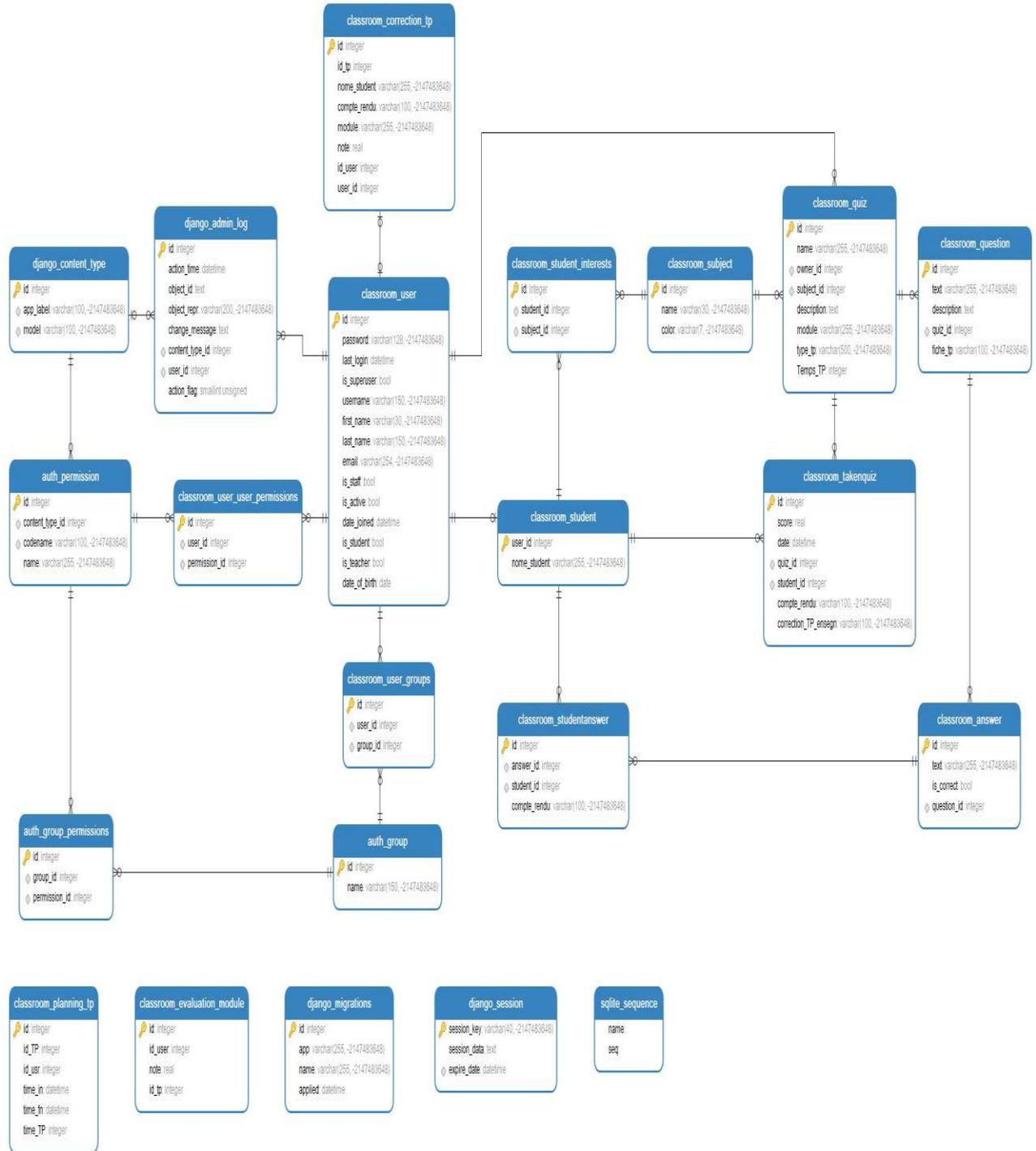


Figure III.26 : Structure de la base de données

III.4.3.9 Architecture réseau du TP réalisé

La figure ci-dessous résume la topologie et l'architecture réseau du travail pratique qu'on a réalisé.

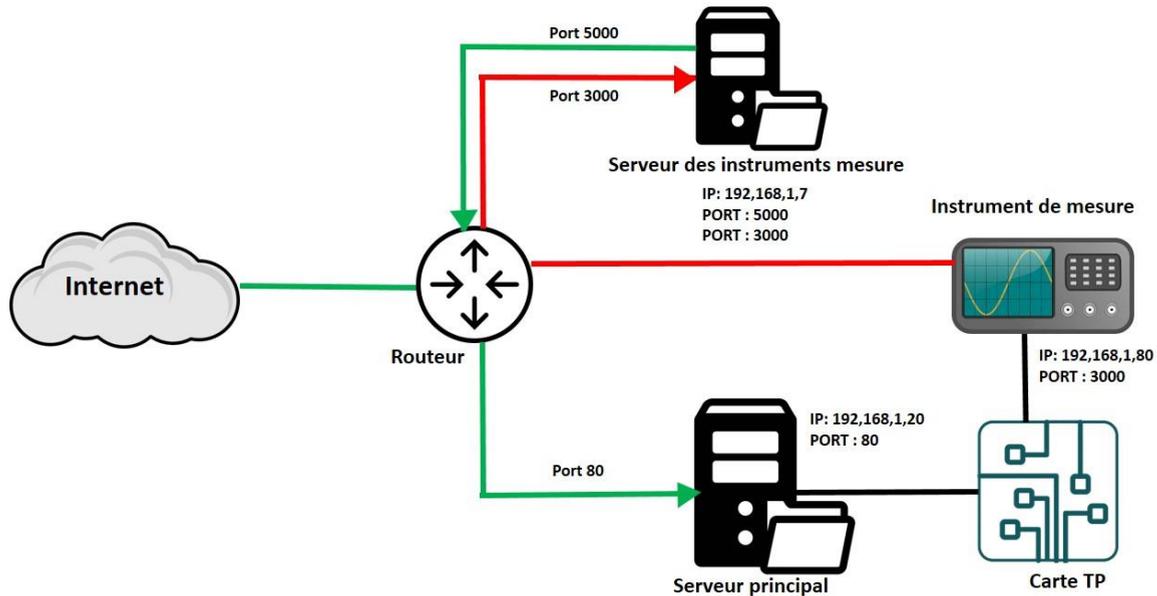


Figure III.27: Architecture réseau du TP réalisé.

Toutes les adresses sont attribuées comme statiques. Ensuite, nous configurons le routeur, sachant que .

- Le serveur principal fonctionne sur le port 80
- Le serveur instrument de mesure fonctionne sur le port 5000
- Pour instruments de mesure. nous avons utilisé le port 3000 pour afficher l'écran sur le logiciel utilisé dans le serveur de l'instrument
- Le système fonctionnera quel que soit réseaux LAN ou WAN

Lorsqu'un utilisateur se connecte à l'adresse du routeur (ip public). Il est dirigé vers le serveur principal

III.5 Résultats, discussions et perspectives

III.5.1 Présentation de l'application

La succession du processus de connexion est comme suit

page d'accueil de l'application : Affiche des informations sur le projet et le fonctionnement de l'application

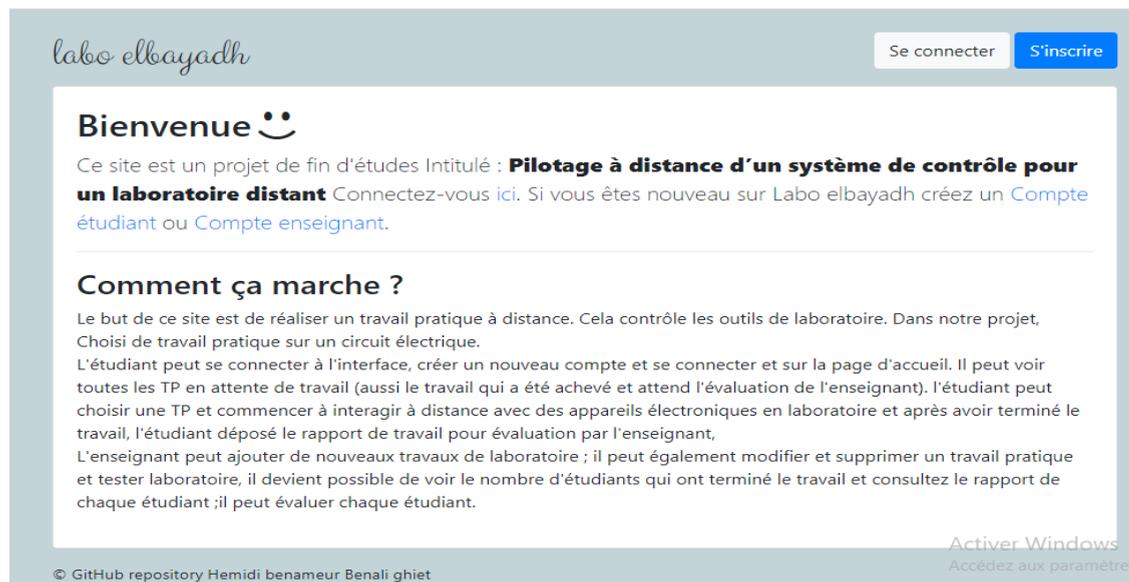


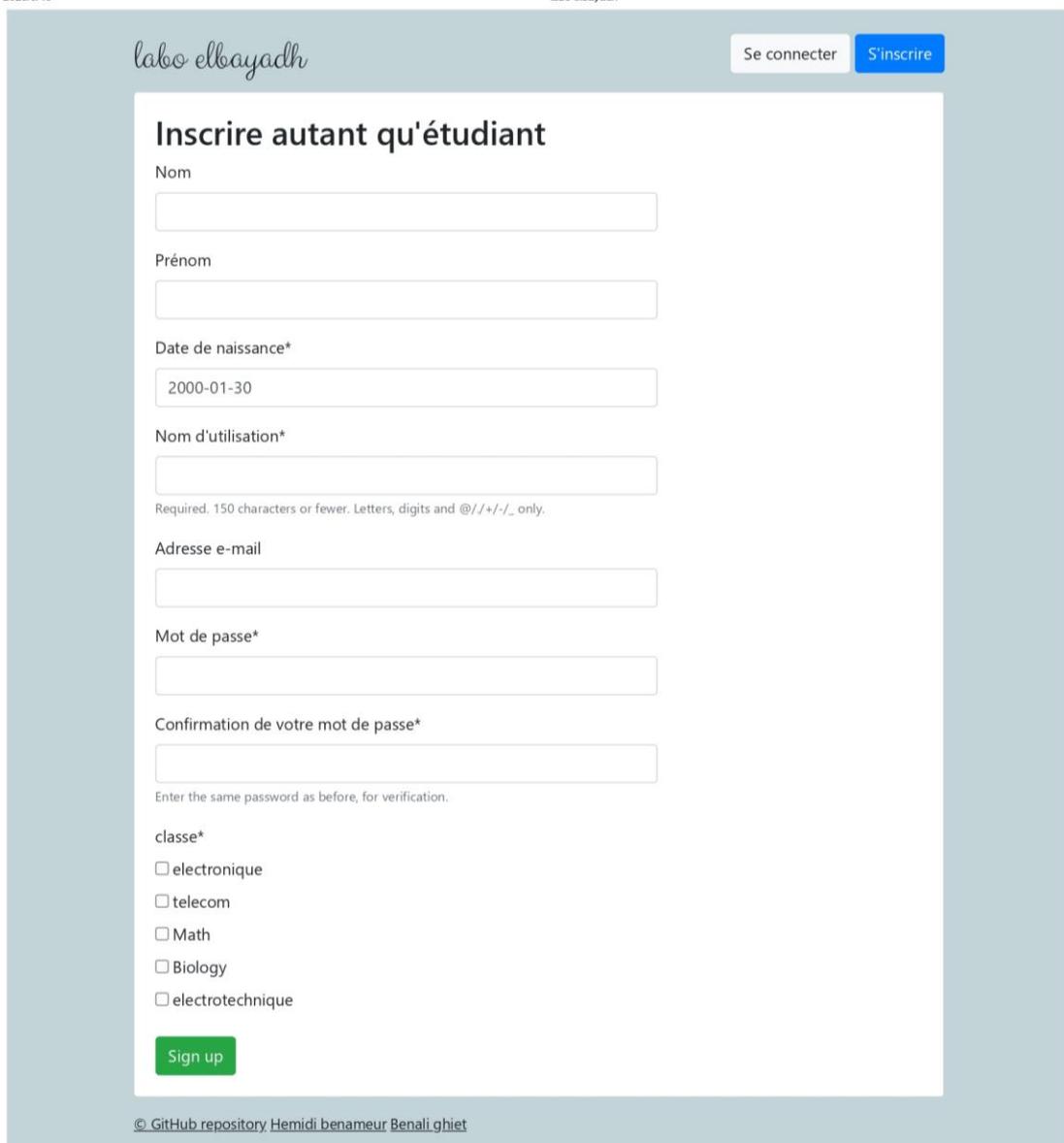
Figure III.28: page d'accueil de l'application

Choisir le type de compte : Enseignant ou étudiant



Figure III.29: Inscription de l'utilisateur, choix du type de compte

Interface inscription des étudiants : cette page permet aux étudiants de s'inscrire sur la plateforme



The screenshot shows a web interface for 'labo elbayadh'. At the top right, there are two buttons: 'Se connecter' (white) and 'S'inscrire' (blue). The main heading is 'Inscrire autant qu'étudiant'. The form includes the following fields and options:

- Nom: Text input field.
- Prénom: Text input field.
- Date de naissance*: Text input field containing '2000-01-30'.
- Nom d'utilisation*: Text input field with a note: 'Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.'
- Adresse e-mail: Text input field.
- Mot de passe*: Text input field.
- Confirmation de votre mot de passe*: Text input field with a note: 'Enter the same password as before, for verification.'
- classe*: A list of radio button options:
 - électronique
 - telecom
 - Math
 - Biology
 - electrotechnique

At the bottom of the form is a green 'Sign up' button. The footer of the page contains the text: '© GitHub repository Hemidi benameur Benali ghiet'.

Figure III.30:Interface inscription des étudiants

Interface inscription de l'enseignant : cette page permet aux enseignants de s'inscrire sur la plateforme (figure III.31)

labo elbayadh

Se connecter S'inscrire

Inscrire autant qu'enseignant

Nom

Prénom

Date de naissance*

Nom d'utilisation*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Adresse e-mail

Mot de passe*

Confirmation de votre mot de passe*

Enter the same password as before, for verification.

Sign up

© GitHub repository Hemidi benameur Benali ghiet

Act
Accé

Figure III.31 : Interface inscription de l'enseignant

Page d'accueil de l'enseignant: affiche le travail pratique qui a été ajouté par l'enseignant.

labo elbayadh

Connecté en tant que **hemidi**. [Se déconnecter.](#)

Bienvenue, monsieur **hemidi benameur** Voici une liste de toutes les TP que vous avez ajoutées. Pour modifier ou supprimer cliquez sur le titre de TP. Pour ajouter un nouveau TP, cliquez sur Ajouter un TP. Pour voir les résultats et les rapports des étudiants, cliquez sur Afficher les résultats à côté de chaque TP

Ajouter un TP

Title TP	Classe	N qui a terminé
Redresser et filtrage	electronique	4

[voir le TP](#) [Résultats](#)

© GitHub repository Hemidi benameur Benali ghiet

Figure III.32 :Page d'accueil de l'enseignant

Interface ajouter un nouveau travail pratique TP (enseignant)

labo elbayadh  Connecté en tant que **hemidi**. [Se déconnecter.](#)

[Mon TP](#) / Ajouter un nouveau TP

Ajouter un nouveau TP

Titre TP*

Les filières*

Type de circuit électrique*

Matière*

Temps de TP min*

L interval temporel de calendrier (À l'heure)*

Active le calendrier

Envoyer des notifications au étudiants par email

© GitHub repository Hemidi.benameur.Benali.ghiet

Figure III.33 : Interface de création d'un nouveau travail pratique TP

Interface ajouter une description et un fichier compte-rendu (enseignant)

labo elbayadh  Connecté en tant que **hemidi**. [Se déconnecter.](#)

[mon TP](#) / [Titre TP](#) / informations TP

informations TP

Veillez choisir un fichier PDF ou DOCX.

Compte-rendu*

Description de TP *

Activer Wi
Accédez aux p

Figure III.34 : Interface de dépôt du protocole du TP (enseignant)

Interface résultat : affiche la liste des étudiants qui ont déjà effectué le TP Et le calendrier pour ça TP

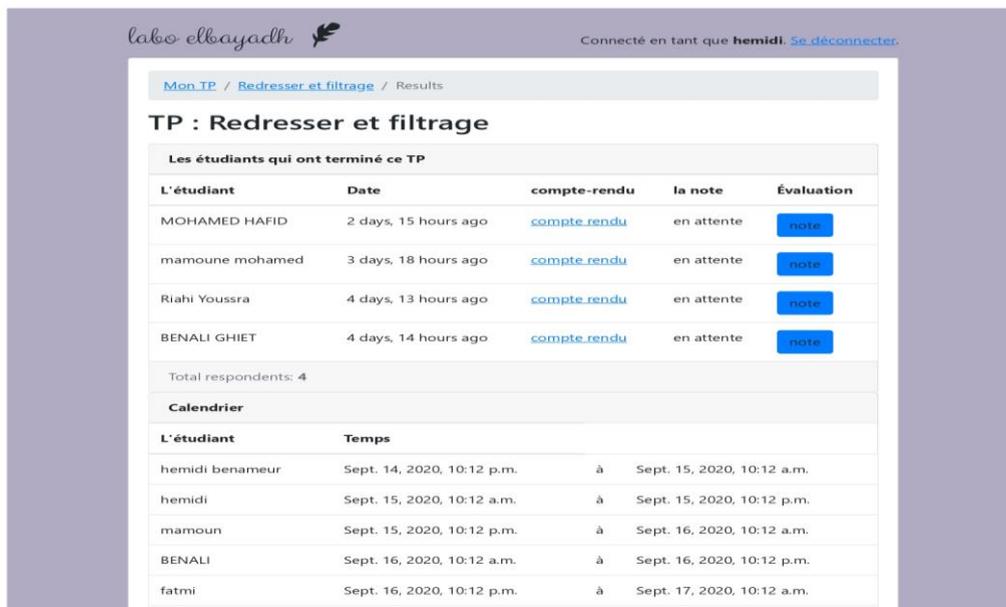


Figure III.35 : Interface résultats du TPs

Interface évaluation les étudiants (enseignant)Et envoyer la correction



Figure III.36 : Interface évaluation des étudiants

La page d'accueil de l'étudiant: affiche le travail pratique en attente de fin

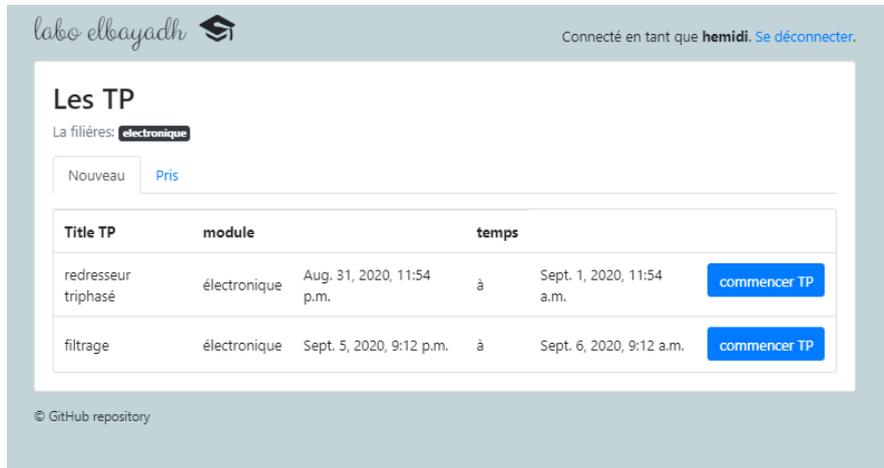


Figure III.37: page d'accueil de l'étudiant

Annonces de TP (étudiants)



Figure III.38 : Annonces de TP (étudiants)

Interface réalisation de TP (étudiants)

labo elbayadh
Connecté en tant que **hemmmmm**. [Se déconnecter.](#)

le temps restant 0d 0h 3m 5s

Title de TP
filtrage

Description
exemple de description TP

Cliquez [ici](#) pour voir le compte-rendu

Oscilloscope

rappel !
Après avoir terminé le travail, **déposé le compte-rendu**. Appuyez ensuite sur confirmer pour l'enseignant évaluer le TP.
Lorsque vous avez terminé ce processus, vous ne pouvez pas revenir à cette page. Vous devez vous assurer que le travail est bien fait

Compte rendu*

لم يتم اختيار أي ملف اختيار ملف

[confirmer](#)

© GitHub repository Hemidi.benameur Benali.ghiet

Figure III.39 : Interface réalisation de TP (étudiants)

Interface de l'administrateur

Grâce à cette interface, l'administrateur peut contrôler tous les utilisateurs et gère les droits sur le système

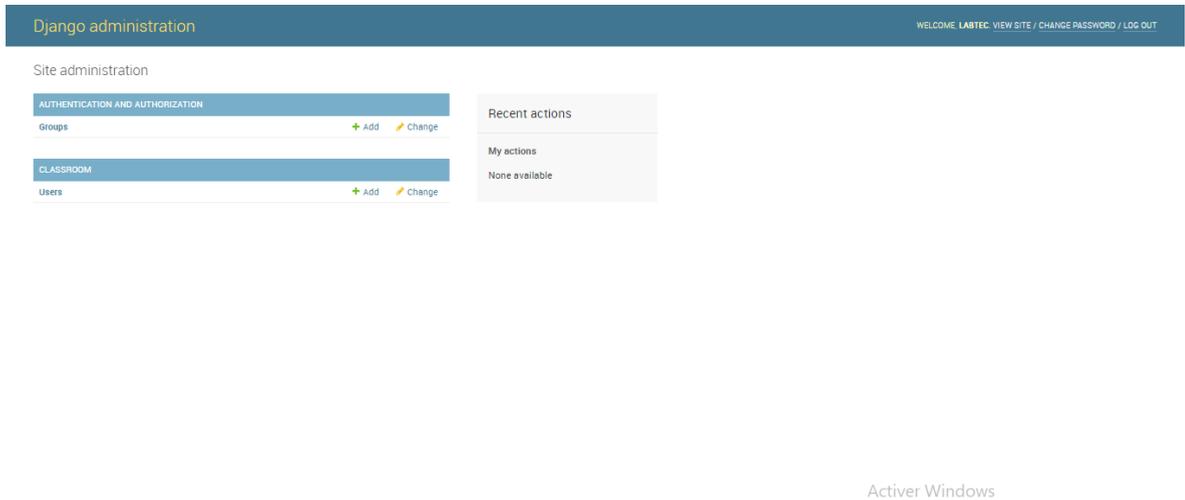


Figure III.40 : Interface de l'administrateur (page d'accueil)

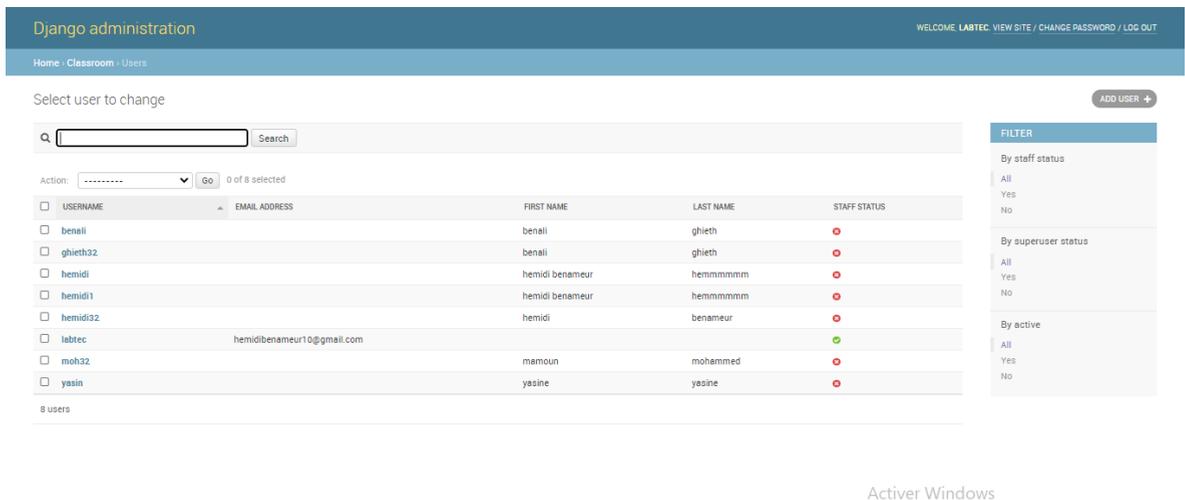


Figure III.41 : Interface de l'administrateur (Liste des utilisateurs)

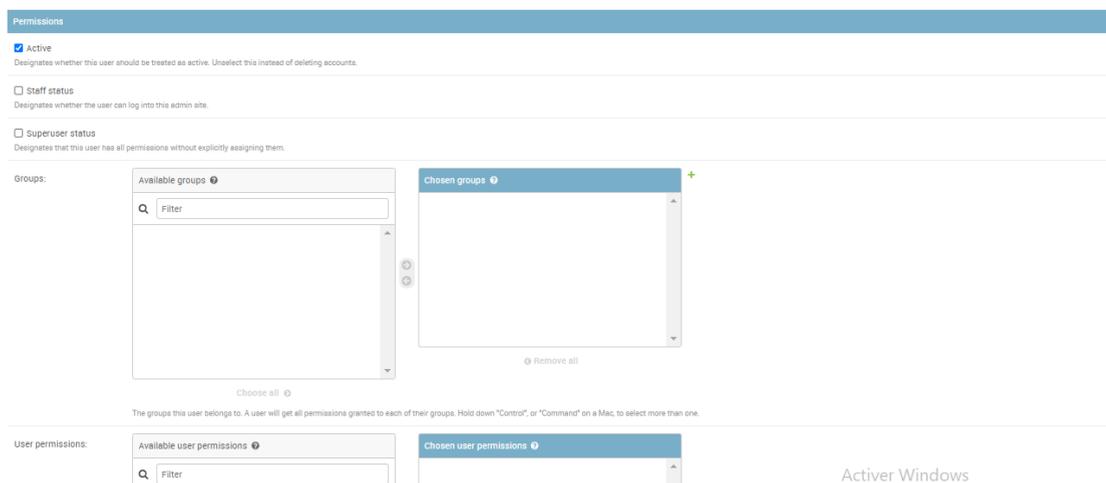


Figure III.42 : Interface de l'administrateur (Autorisations)

III.5.2 discussions et perspectives

Cette réalisation a été succédée par plusieurs tests afin de ressortir les points forts et les points faibles de notre projet à savoir:

a) Les points forts

- **Le temps de réalisation TP**

Le temps nécessaire pour terminer les travaux pratiques est très court par rapport au laboratoire réel. La plupart des étudiants qui fait le test sont effectuent des travaux pratiques en 10 à 20 minutes. Sachant qu'un tel test peut prendre une à deux heures dans laboratoire réel.

- **Vitesse d'application et l'interface**

L'interface de l'application est facile à utiliser et rapide. Tous les étudiants ont été évalués avec un score complet

- **Erreurs et problèmes**

Aucune erreur ou problème n'était enregistré pendant le test par les étudiants

- **Des dossiers**

Notre system supporté n'importe quel type de fichier PDF ou DOCX pour ajouter ou téléchargé

- **Site sécurisé**

Pour protéger tout site Web du piratage, il est important de connaître les risques. Nous avons suivi les conseils du site officiel de Django via le lien suivant[43]

b) les points faibles

- **Il ne prend pas en charge android et IOS(iPhone OS)**

Il y a une difficulté de faire le TP par téléphone .Que ce soit sur Android ou IOS. à cause que le gabarit ne prend pas en charge les dimensions de l'écran du téléphone mobile

Pour résoudre ce problème, nous avons commencé à développer une application prend Android et IOS utilisant environnement 'Flutter'. et Langage de programmation Dart.

Pour partager le donné entre serveur et l'application utilisé technologie API (Application Programming Interface) a l'aide de bibliothèque 'Django REST Framework'

- **Problèmes de temps**

Aucune étudiant réclamer sur le temps de réponse du système mais il y a deux contrainte temporelle importants sont :

. Le temps nécessaire pour envoyer les données du l'oscilloscope à serveur est retardé d'environ une à deux secondes. Bien sûr, la plupart des étudiants ne l'ont pas remarqué, mais il y a eu un retard dans l'envoi des données .

. Lors de l'ajout d'un nouveau TP par l'enseignant, et après l'ajout de la fonction d'envoi de notifications à l'e-mail, nous avons remarqué un certain retard dans la création des travaux pratiques, environ 2-3 secondes. Ceci est dû à la raison pour laquelle l'application envoie un e-mail aux étudiants. Le délai dépend de la connexion Internet et du nombre d'étudiants auxquels un message est envoyé par e-mail.

III.6 Conclusion

Le présent chapitre a décrit les différentes étapes de réalisation de notre projet. En commençant par la partie modélisation qui nous a permis de connaître les différents acteurs du système, leurs relations et leurs interactions. Les diagrammes UML développés nous ont aussi servi pour énumérer les différents paquetages à développer. Comme a été expliqué, plusieurs solutions ont été proposées mais nous avons opté par la solution la plus optimale et homogène, notamment pour éviter des outils hétérogènes dans la partie Back-end qui n'ajoutent que de la complexité et des failles de sécurité.

Conclusion Générale

Le travail présenté dans ce mémoire a été consacré à la mise en œuvre d'un environnement de travaux pratique à distance via le réseau Internet dans le domaine électronique.

Pour mettre en œuvre un TP à distance, nous étions chargés de mettre en place une architecture matérielle spécifique et de développer une application web performante et facile à utiliser permettant à un étudiant distant de réaliser un TP à distance.

Réaliser un TP à distance dans un environnement d'apprentissage en ligne LMS renforce les activités d'apprentissage. Dans un laboratoire classique, les étudiants en avoir à répéter les expériences et le temps est souvent limité, alors que dans la nouvelle situation, les étudiants peuvent travailler dans des laboratoires distants 24/7, de sorte que la possibilité d'accéder est plus grande que dans les laboratoires locaux.

Nous avons essayé dans notre projet de proposer une solution simple, ergonomique et évolutive, facile à utiliser par tous les acteurs du système.

Ce projet nous a offert l'occasion d'approfondir nos connaissances en matière de modélisation à travers l'utilisation d'UML. De plus, nous nous sommes familiarisés avec les différents outils que nous avons employés à savoir Django et flask et l'ensemble des outils de programmation web tels que bootstrap, JavaScript et HTML5, SVG, AJAX, qui constituait la prenante partie de notre application.

Le développement de notre application, nous a également permis d'approfondir nos connaissances en matière de conduite de projet et de le mettre en pratique. Certes, le travail effectué était tout au long de sa réalisation une réponse à ce qui était proposé par le thème du sujet. Mais, il peut être amélioré de différentes manières et évoluer vers une solution plus optimisée.

Notre application reste toujours extensible par d'autres développeurs pour d'autres améliorations et ajout de nouvelles fonctionnalités ou encore pour l'entretenir et l'adapter aux besoins émergents, notamment les nouvelles technologies de communications mobiles

Enfin, ce mémoire n'est qu'un départ et un travail modeste pour des futures contributions et il reste encore beaucoup de perspectives à explorer, telles que :

- Créez une application Android et associez-la au projet à l'aide de API (djangorestframework).Au moment où nous écrivons les lignes, l'application est encore en développement. Vous pouvez vérifier les mises à jour dans<https://github.com/hemid32>
- Le projet doit ajouter des nombreuses fonctionnalités telles que le chat entre les étudiants. Ce qui améliore le travail collectif
- Jusqu'à présent, nous utilisons un circuit pré-préparé en laboratoire. Nous espérons au futur à rendre la matière plus flexible, afin que l'étudiant puisse créer un circuit entièrement électrique à l'intérieur du site et qu'il puisse apporter librement des modifications

Bibliographie

- [1]. Ramdane, M « Un environnement de travail collaboratif dédié aux Travaux Pratiques à distance ».Mémoire Magistère Université Mouloud Mammeri Tizi-ouzou ,2001
- [2]. NEAU, N. ArgoGraph: Un support au débat scientifique dans le cadre de travaux pratiques pour l'apprentissage des sciences expérimentales. Thèse de doctorat en informatique. Le Mans: Université du Maine, 2003, 287 p
- [3]. Cooper M « Remote controlled experiments for teaching over the Internet : a comparison of approaches developed in the PEARL Project». Conférence ASCILITE (Australian Society for Computers in Learning in Tertiary Education), Auckland, Nouvelle Zélande, du 08 au 12 Décembre 2002.
- [4]. Gravier C « Vers la généralisation de manipulations distantes et collaboratives d'instruments de haute technologie ». Thèse de Doctorat, paris 2007.
- [5]. Djamila , M « plate-forme pour les travaux pratiques à distance sur le Web ». Thèse de Doctorat UFAS 2012. P 14-20
- [6]. D. K. Cohen, (1990), "Revolution in one classroom: The case of Mrs. Oublier," Educ.Eval. Policy Anal., vol. 12, no. 3, pp. 311–329
- [7]. J. L. Polman, (2000), "Designing Project-Based Science: Connecting Learners Through Guided Inquiry (Ways of Knowing in Science Series)".
- [8]. A. Hofstein and V. N. Lunetta, (1982), "The Role of the Laboratory in Science Teaching: Neglected Aspects of Research," Rev. Educ. Res., vol. 52, no. 2, pp. 201–217
- [9]. Lazarowitz, R., & Tamir, P. (1994). Research on using laboratory instruction in science. Handbook of research on science teaching and learning, 94-130.
- [10]. Mostefaou . H «Conception de laboratoire d'électronique accessible à distance ». Thèse de Doctorat mostaganem .2018.p 25
- [11]. F. Reader, F. S. Company, F. E. Only, J. Garc, U. Hern, U. Hernandez-Jayo, J. García- Zubia, I. Angulo, D. Lopez-de-Ipiña, P. Orduña, J. Irurzun, and O. Dziabenko,(2010), "LXI Technologies for Remote Labs: An Extension of the VISIR Project," Int. J. Online Eng., vol. 6, no. S11, pp. 160–170

- [12]. Harward, V. J., Del Alamo, J. A., Lerman, S. R., Bailey, P. H., Carpenter, J., DeLong, K., ...& Long, P. D. (2008). The ilab shared architecture: A web services infrastructure to build communities of internet accessible laboratories. *Proceedings of the IEEE*, 96(6), 931-950.
- [13]. “<https://icampus.mit.edu/projects/ilabs/>.” . Consulté le 22/11/2017.
- [14]. National Instruments, , (2009) “NI ELVIS II Series Specifications,” <Http://Www.Ni.Com/Pdf/Manuals/372590B.Pdf>, p. 11.
- [15]. Tawfik, M., Sancristobal, E., Martin, S., Gil, R., Diaz, G., Colmenar, A., ...& Hakansson . L (2013). Virtual instrument systems in reality (VISIR) for remote wiring and measurement of electronic circuits on breadboard. *IEEE Transactions on Learning Technologies*, 6(1), 60-72.
- [16]. Tawfik, M., Salzman, C., Gillet, D., Lowe, D., Saliyah-Hassane, H., Sancristobal, E., & Castro . M (2014, February). Laboratory as a Service (LaaS): A model for developing and implementing remote laboratories as modular components. In *Remote Engineering and Virtual Instrumentation (REV), 2014 11th International Conference on* (pp. 11-20). IEEE
- [17]. P. Sant, (2003), “Formulation des questions,” pp. 1–7,. Voir l’adresse https://www.quint-essenz.ch/fr/public_assets/62/download
- [18]. Bagnasco, A., & Scapolla, A. M. (2003, March). A Grid of Remote Laboratory for Teaching Electronics. In *LeGE-WG 2, Eur. 2*, pp. 1–5,
- [19]. Scapolla, A. M., Bagnasco, A., Ponta, D., & Parodi, G. (2005). A modular and extensible remote electronic laboratory. *International Journal of Online Engineering (iJOE)*, 1(1)..
- [20]. J. Machotka and Z. Nedic, (2009), “Remote laboratory to attract engineering students.” (July 2009) in *37th SEFI Conference*, Netherlands
- [21]. Machotka, J., & Nedic, Z. (2006). The remote laboratory NetLab for teaching engineering courses (Thèse doctorat, UICEE).
- [22]. Nafalski, A., Nedic, Z., Machotka, J., Göl, Ö., Scarino, A., Crichton, J., ... & Murray, S. (2009). International Collaboration in Remote Engineering Laboratories: an Approach to Development. *IEEE Transactions on Learning Technologies*, 1-8.

- [23]. Gustavsson, I., Nilsson, K., Zackrisson, J., Garcia-Zubia, J., Hernandez-Jayo, U., Nafalski, A., ...&Lago, T. (2009). On objectives of instructional laboratories, individual assessment, and use of collaborative remote laboratories. *IEEE Transactions on Learning Technologies*, 2(4), 263-274
- [24]. Asumadu, J. A., Tanner, R., Fitzmaurice, J., Kelly, M., Ogunleye, H., Belter, J., &Koh,S. C. (2005). A web-based electrical and electronics remote wiring and measurement laboratory (RwmLAB) instrument. *IEEE Transactions on Instrumentation and Measurement*, 54(1), 38-44
- [25]. J. A. Asumadu and R. Tanner,(2001),“Remote wiring and measurement laboratory,” *Proceedings of the 2001 American Society for Engineering Education Annual Conference & Exposition*.Vol.6,p.1.
- [26]. Matt Richardson et Shawn Wallace, À LA DÉCOUVERTE DU Raspberry Pi
- [27]. https://lealinux.org/documentations/Pr%C3%A9sentation_du_Raspberry_Pi24/07/2020
- [28]. <https://yadom.fr/raspberry-pi-4-model-b-1gb.html>24/07/2020
- [29]. J. Pieper, “Volume 1 : Syntax and Style,” vol. 1, 1999.
<http://www.ivifoundation.org/docs/scpi-99.pdf>
- [30]. A. Function and P. Generators, “TGxx11 Series Function / Arbitrary / Pulse generators: <https://www.butterfly.com/PDF/.../AimTTi-TG5011-TG2511.pdf>
- [31]. Site officiel python Mis à jour le sept. 16, 2020<https://docs.python.org/fr/3/tutorial/>20/07/2020
- [32]. Site officiel python Mis à jour le sept. 16, 2020
<https://docs.python.org/fr/2/library/subprocess.html>16/08/2020
- [33]. <http://www.additeam.com/SSII/javascript/> , 25/08/2020
- [34]. site python doctor<https://python.doctor/page-django-introduction-python> ,18/07/2020
- [35]. Gordon,the GPIO utility <http://wiringpi.com/the-gpio-utility/>25/07/2020
- [36]. Tammy Noergaard , *Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers (Embedded Technology)* , ELSEVIER , feb 28 , 2005

- [37]. James, P. V. (1999). Report of the Expert Meeting on Virtual Laboratories. Organized by the International Institute of Theoretical and Applied Physics (IITAP), Ames, Iowa 10-12 May, with support of the United Nations Educational, Scientific and Cultural Organization
- [38]. Alexiou, A., Bouras, C., Giannaka, E. (2004). Virtual Laboratories In Education: A cheap way for schools to obtain laboratories for all courses by using the Computer Laboratory. 18th IFIP WCC – Workshop Tel'04, Technology Enhanced Learning, Toulouse, France, 22-27 Aug, pp. 19-28
- [39]. Saliah-Hassan, H. (2011). Laboratoires en ligne. http://profetic.org/dossiers/article.php3?id_article=907. (Consulté en 2011).
- [40]. J. Garcia-Zubia U. Hernández-Jayo, LXI technologies for remote labs: an extension of the VISIR project, Faculty of Engineering, University of Deusto, Bilbao, Spain, 29 June - 2 July, 2010
- [41]. Richard Marr, Is it possible to use Django and Node.js? ,
« <https://stackoverflow.com/questions/13795313/is-it-possible-to-use-django-and-node-js> » Dec 10 '12 at 11:06
- [42]. Hemidi, B., Benali, G. https://github.com/hemid32/flask_lab 24/09/2020
- [43]. Site officiel Django <https://docs.djangoproject.com/fr/3.0/topics/security/> 27/07/2020
- [44]. A. MEDJAHED K. LARBAOUI « Le laboratoire de travaux pratiques à distance d'électronique, Mémoire master, UM, 2014

Annexe A

Django avec Apache2 et mod_wsgi

1. configuration fichier 000-default.conf

Ouvrez un terminal et tapez la commande suivante

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

Modifiez le fichier comme suit

```
<VirtualHost *:80>
ServerAdmin webmaster@localhost
Alias /static [path_to_static_folder]

<Directory [path_to_static_folder]>
Require all granted
</Directory>

<Directory [path_to_project]>
<Files wsgi.py>
    Require all granted
</Files>
</Directory>

WSGIDaemonProcess [name] python-home=[path_to_virtualenv] python-path=[path_to_project]
WSGIProcessGroup [name]
WSGIScriptAlias / [path_to_project_wsgi.py]

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

</VirtualHost>
```

1. configuration fichier wsgi.py

Modifiez le fichier wsgi.py comme suit

```
1 import os
2 import sys
3 import site
4 # path_to_lib_python
5 site.addsitedir('/root/Desktop/testdjangoschool/lib/python3.x/site
    -packages')
6 #path_to_app
7 import sys
8 sys.path.append('/root/Desktop/testdjangoschool/src/django_school
    /django_school')
9 #path_to_project
10 sys.path.append('/root/Desktop/testdjangoschool/src/django_school')
11 from django.core.wsgi import get_wsgi_application
12 #os.environ.setdefault("DJANGO_SETTINGS_MODULE", "django_school
    .settings")
13 os.environ['DJANGO_SETTINGS_MODULE'] = 'django_school.settings'
14 application = get_wsgi_application()
```

Un guide pour résoudre les problèmes que vous pouvez rencontrer dans ce fichier https://github.com/hemid32/django_labo/blob/master/apache2_config

Observation

Vous pouvez exécuter le serveur django et lier à l'adresse publique sans utiliser Apache en exécutant la commande suivante

```
python3 manage.py runserver 0.0.0.0 :80
```

Annexe B

Contributions :

a) Les exigences d'exploitation :

Pour serveur 1 (django) :

- 1 – python3.X
 - 2- installe python-pip
 - 3- installe les bibliothèques suivantes
- virtualenv
requests
django==2.2.1
django-crispy-forms==1.7.0
pytz==2017.3

pour serveur 2 (Flask) :

- 1 – python2.x
 - 2- installe python-pip
 - 3- installe les bibliothèques suivantes
- pillow==4.2.0
pyscreenshot==0.4.2
flask==0.12.2

b) Git et GitHub

Nous utilisons git comme système de contrôle de version pour notre base de code. Si vous n'avez jamais utilisé git ou un DVCS similaire (ou même un VCS) auparavant, nous vous suggérons fortement de jeter un œil à l'excellente documentation disponible pour git en ligne. Le livre de la communauté Git ou les vidéos Git sont tous deux d'excellents moyens d'apprendre git. Faites-nous confiance quand nous disons que git est un excellent outil. Cela peut sembler intimidant au début, mais après un certain temps, vous l'aimerez (espérons-le) autant que nous. Cependant, vous enseigner git dépasse largement le cadre de ce document.

- 1 - Log in to GitHub

2- Créez un fork du django_lab en cliquant sur le bouton fork.

3- Clonez votre fork de notre -repository- sur votre ordinateur. Votre fork aura le nom git remote "origin" et vous serez sur la branche "master":

```
git clone https://github.com/hemid32/django\_lab
```

4- Ajouter le repo django_lab en tant que source distante

```
git remote add django_lab https://github.com/hemid32/django\_lab
```

Désormais, chaque fois que vous souhaitez créer un patch, suivez les étapes suivantes:

5- Créez une nouvelle branche nommée de manière appropriée dans votre référentiel local pour cette fonctionnalité ou correction de bogue spécifique. (En conservant une nouvelle branche par fonctionnalité, nous pouvons facilement intégrer vos modifications sans extraire d'autres éléments qui ne sont pas censés être extraits.):

```
git checkout -b new_feature
```

6-Modifiez le code pour faire ce que vous voulez (par exemple, corrigez-le).

7-Testez le code. Essayez de faire cela même pour de petites corrections. Vous ne savez jamais si vous avez introduit un bug étrange sans test.

8-Donnez à chaque commit un message de commit approprié, afin que les autres qui ne sont pas familiers avec le sujet aient une bonne idée de ce que vous avez changé.

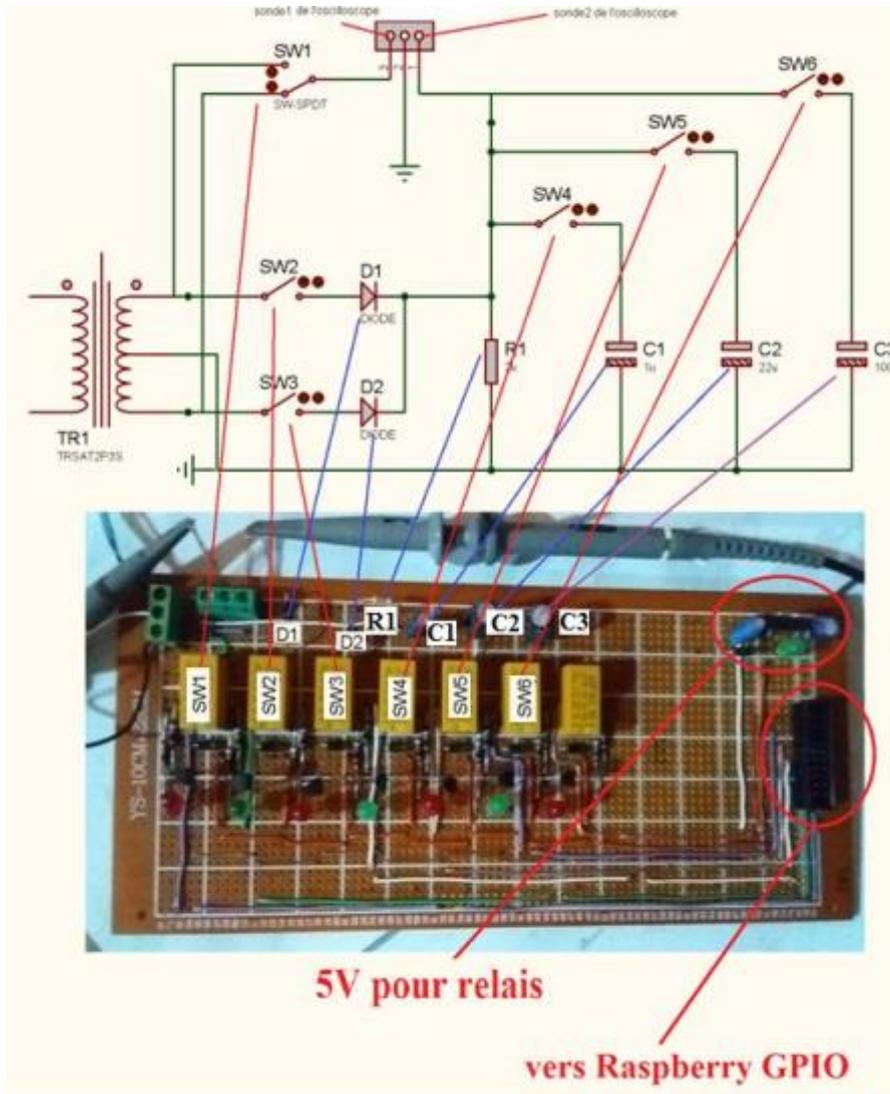
9-Une fois que vous êtes satisfait de vos modifications, extrayez notre référentiel en amont et fusionnez-le avec votre référentiel local. Nous pouvons extraire vos données, mais comme vous savez exactement ce qui a changé, vous devez effectuer la fusion:

```
git pull django_lab master
```

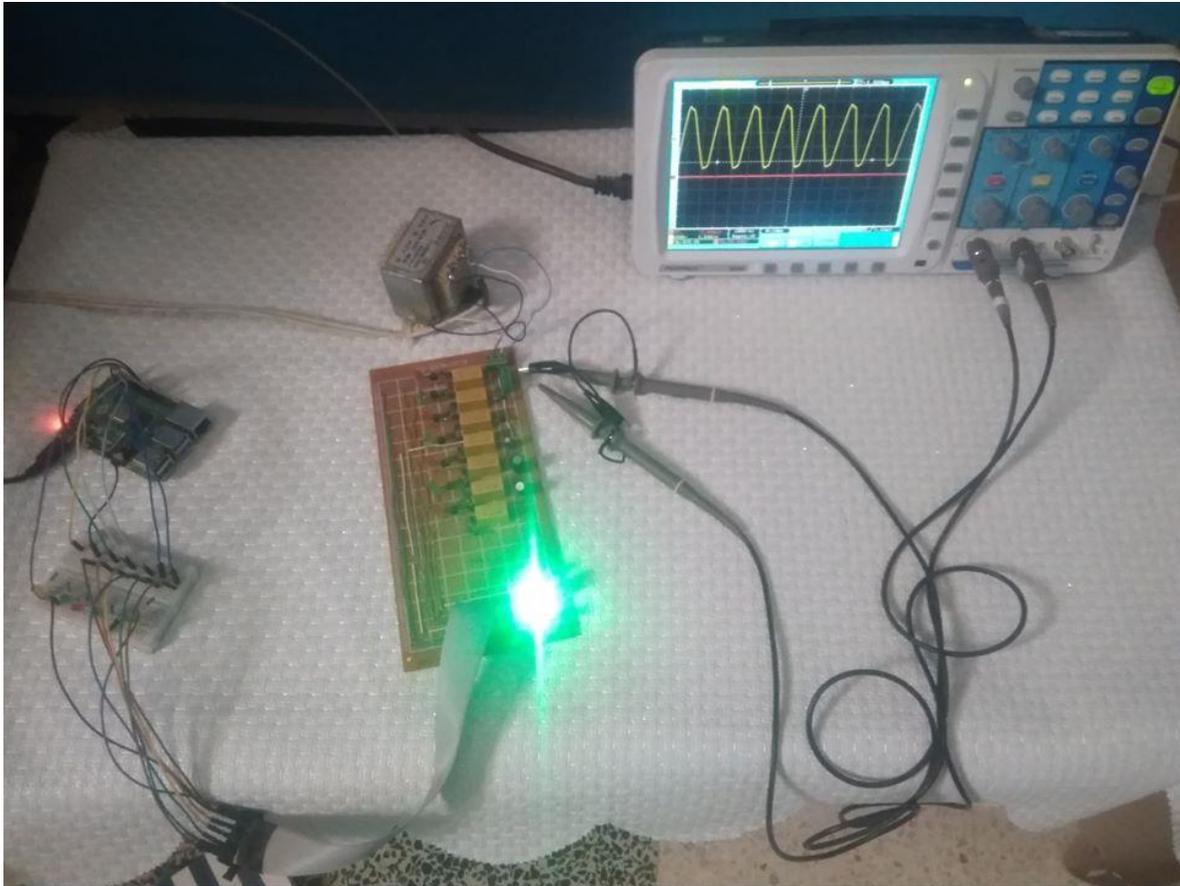
10-Poussez (push) votre branche locale dans votre référentiel distant sur GitHub:

```
git push origin new_feature
```

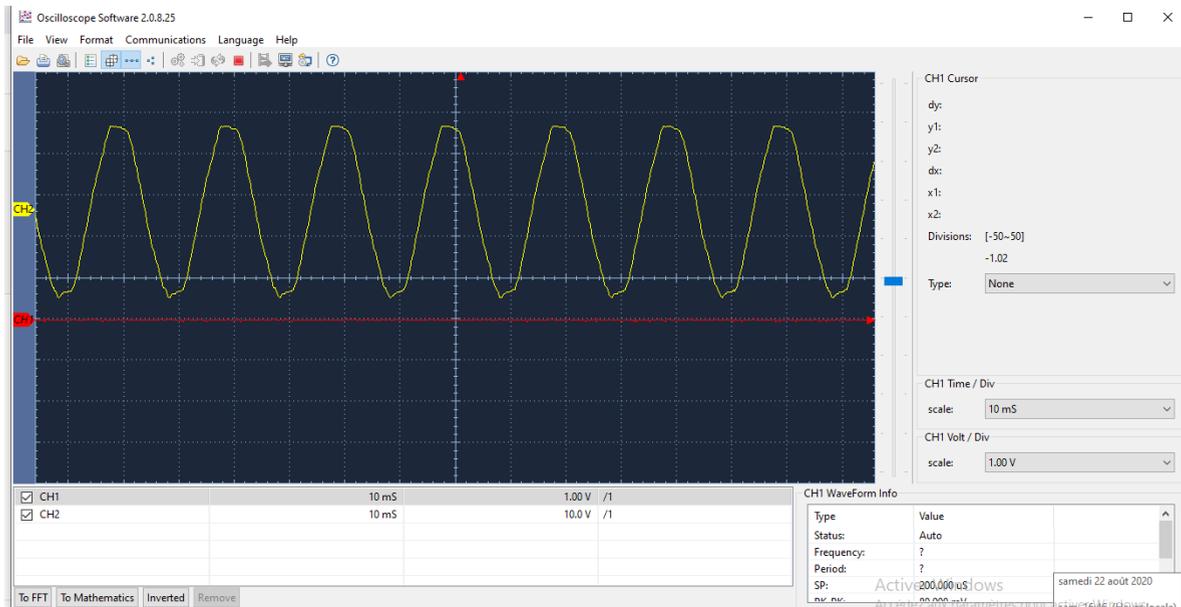
Annexe C



Carte TP



réalisation de projet



Écran du serveur de l'instrument de mesure (flask) qui partage sur le port 5000

Abstract:

Along with the Corona outbreak that rocked the world, e-learning platforms have emerged to replace and solve problems with traditional systems, especially labs. The main objective of this thesis is to develop a new system for the remote control of oscilloscope measuring devices and electronic circuits for performing experiments via the Internet browser.

The system we are about to introduce must be easy to control, easy to maintain, flexible to program and use, and must be scalable with precision and power and the ability

to support many users. It also requires reducing system development costs by using free and 'open source' resources. The system must also be resembling the real lab.

We therefore need a physical architecture and special software, using the 'Raspberry Pi' platform which has been linked to the Internet with a 'web' server. In the development of this server, we mainly base on the modern programming language 'Python' using the development environment 'django' in addition to the well known 'Web' technologies such as 'HTML', 'JavaScript', 'AJAX'... etc.

Finally, this project will allow us to create a distance learning system that simulates laboratories in universities, which is of economic and scientific interest.

Keywords: Remote laboratory. Python. website . Remote control . Electronic. Embedded systems. Distance Learning ,django , flask

ملخص :

تزامنا و جائحة كورونا التي هزت العالم بجميع أركانه برزت منصات التعليم الإلكتروني على مستوى العالم لاستبدال وحل المشكلات المتعلقة بالأنظمة التقليدية، وخاصة المختبرات. الهدف الرئيسي من هذه المذكرة هو تطوير نظام جديد للتحكم والتوصيل عن بعد أجهزة القياس مثل راسم الذبذبات والدارات الإلكترونية لإجراء تجارب عبر متصفح الانترنت النظام الذي نحن بصدد طرحه يجب ان يتميز بسهولة التحكم و سهولة الصيانة و مرونة البرمجة و الاستخدام ، ويكون قابل للتطوير مع السرعة والدقة و إمكانية دعم متعدد المستخدمين ... إلخ . يتطلب أيضًا تقليل تكلفة تطوير النظام باستخدام موارد مجانية ومفتوحة المصدر. يجب ايضا على النظام أن يتيح أكبر قدر من التشابه مع المختبر الحقيقي. لذلك نحتاج بنية مادية وبرمجية خاصة، وذلك باستخدام منصة راسبيري التي تم ربطها بالإنترنت باستخدام خادم ويب . نعتمد كليا في تطوير هذا الخادم على لغة برمجة حديثة بايثون باستعمال بيئة التطوير دجانغو بالإضافة الي تقنيات الويب المعروفة مثل اش تي ام ال و جافا سكريبت و اجاكس ... الخ

و في النهاية يتيح لنا هذا المشروع إنشاء نظام تدريس عن بعد يحاكي المختبرات في الجامعة مما يعود بنفع اقتصادي و علمي .

كلمات مفتاحية:

مختبر بعيد . بايثون . موقع ويب . تحكم عن بعد . إلكترونيك . أنظمة مضمنة . التعليم عن بعد

Résumé :

Parallèlement à l'épidémie de Corona qui a secoué le monde, des plateformes d'apprentissage en ligne sont apparues pour remplacer et résoudre les problèmes liés aux systèmes traditionnels, en particulier les laboratoires. L'objectif principal de cet mémoire est de développer un nouveau système de commande à distance des appareils de mesures d'oscilloscope et les circuits électroniques pour réaliser des expériences via le navigateur Internet.

Le système que nous sommes sur le point d'introduire doit être facile à contrôler, facile à entretenir, flexible à programmer et à utiliser, et doit être évolutif avec la précision et la puissance et la possibilité.

De supporter des nombreux d'utilisateurs . Cela nécessite également de réduire les coûts de développement du système en utilisant des ressources gratuites et 'open source'. Le système doit également être. Ressemblant au laboratoire réel.

Nous avons donc besoin d'une architecture physique et de logiciel spécial, utilisant la plateforme 'Raspberry_Pi' qui a été reliée à Internet avec un serveur 'Web'. Dans le développement de ce serveur, on base surtout sur le langage de programmation moderne 'Python' en utilisant l'environnement de développement 'django' en plus des technologies 'Web' bien connues telles que 'HTML', 'JavaScript', 'AJAX' ... etc.

Enfin ce projet nous permettra à réaliser un système d'apprentissage à distance simule les laboratoires dans les universités ce qui présente un intérêt économique et scientifique

Mots clés: Laboratoire à distance. Python. site Internet . Télécommande . Électronique. Systèmes embarqués. L'enseignement à distance , django ;