



École Polytechnique de l'Université de Tours  
 64, Avenue Jean Portalis  
 37200 TOURS, FRANCE  
 Tél. +33 (0)2 47 36 14 14  
[www.polytech.univ-tours.fr](http://www.polytech.univ-tours.fr)

## Spécialité Informatique Industrielle

### Version : 02 10/11/19

<b>CAHIER DE SPECIFICATION &amp; PLAN DE DEVELOPPEMENT</b>			
<b>Projet :</b>	Web Based Time Converter (WBTC)		
<b>Emetteur :</b>	J. LOCHE	Coordonnées : jeremy.loche@etu.univ-tours.fr	
<b>Date d'émission :</b>	10/11/19		
<b>Validation</b>			
Nom	Date	Valide (O/N)	Commentaires
<b>Historique des modifications</b>			
Version	Date	Description de la modification	
00	03/10/19	Rédaction initiale du CDS	
01	05/11/19	Soumission pour validation à M.Rolland	
02	10/11/19	Application de corrections demandées par M.Rolland	



## TABLE DES MATIERES

Table des matières .....	3
Cahier de spécification Système .....	5
1. Introduction .....	5
2. Contexte de la réalisation .....	5
2.1. Contexte .....	5
2.2. Objectifs .....	5
2.3. Bases méthodologiques (selon les projets) .....	6
3. Description générale .....	6
3.1. Environnement du projet .....	6
3.2. Caractéristiques des utilisateurs .....	6
3.3. Fonctionnalités et structure générale du système .....	7
3.4. Contraintes de développement, d'exploitation et de maintenance .....	8
4. Description des interfaces externes du logiciel .....	8
4.1. Interfaces matériel/logiciel .....	8
4.2. Interfaces homme/machine .....	9
4.3. Interfaces logiciel/logiciel .....	9
5. Architecture générale du système .....	10
6. Description des fonctionnalités .....	11
7. Conditions de fonctionnement .....	13
7.1. Performances .....	13
7.2. Capacités .....	14
7.3. Modes de fonctionnement (optionnel) .....	14
7.4. Contrôlabilité .....	14
7.5. Sécurité .....	15
7.6. Intégrité .....	15
Plan de développement .....	16
8. Découpage du projet en tâches .....	16
9. Planning .....	20
Glossaire .....	22
Bibliographie .....	23
Index .....	24



## CAHIER DE SPECIFICATION SYSTEME

### 1. Introduction

Ce document présente les spécifications fonctionnelles du projet **Web based time converter (WBTC)**. Il retrace le contexte de sa réalisation, l'environnement (logiciel, matériel et humain) ainsi que les fonctionnalités et l'architecture globale du système.

La **Maîtrise d'ouvrage et l'encadrement (MOA)** seront assurés par Alexis ROLLAND.

La **Maîtrise d'œuvre (MOE)** sera assurée par Jérémy LOCHE.

### 2. Contexte de la réalisation

#### 2.1. Contexte

Le projet Web Based Time Converter s'inscrit au sein **des projets vitrines** de l'école Polytech Tours. Cette réalisation permettra d'illustrer les savoir-faire enseignés par la formation d'informatique industrielle au cours de **salons et de journées portes ouvertes**. Le client est Alexis ROLLAND, professeur au sein de la spécialité informatique industrielle. Les utilisateurs finaux seront les *futurs étudiants* et *visiteurs des manifestations* énoncés précédemment.

#### 2.2. Objectifs

L'objectif de ce projet est de réaliser un dispositif *ludique (et utile)* s'inspirant du célèbre convecteur temporel et de son interface de commande présenté dans les films **Retour vers le futur** (1985).

Le système sera mis en lumière à l'occasion de journées portes ouvertes ou pendant des salons. Il devra donc être robuste et fiable. Deux intérêts sont à dégager de ce projet. Le premier est la mise en œuvre de l'ensemble du spectre de connaissances techniques étudiées en informatique industrielle. Le second est de valoriser ce spectre pendant les manifestations. Il comprend la programmation HDL, analyse et développement électronique, la mise en œuvre de systèmes d'exploitation embarqués et le développement logiciel bas et haut niveau.

Ce projet est une reprise du travail effectué par Florian REMY sur son projet de fin d'étude de 2018-2019. La première édition du projet a mené à la livraison d'études de faisabilités et d'analyses architecturales du système. Cependant, il n'y a pas eu de réalisations matérielles permettant de livrer le système demandé. Toutefois, l'analyse réalisée est un bon point de départ pour cette deuxième édition du projet. On retiendra notamment les points développés par la suite.

La partie matérielle du système devra être composée d'éléments reprenant le plus fidèlement possible les caractéristiques du convecteur temporel et son interface (IHM) de commande.

Cette IHM doit reprendre la première ligne d'afficheurs LED présentée dans les films sur laquelle seront affichées l'heure et la date courante. Le convecteur temporel devra mettre en œuvre des animations lumineuses fidèles à celles du film. L'utilisation de l'espace restant sera à définir. Cette partie libre permettra cependant de mettre en place des éléments plus modernes conservant l'esprit rétro-vintage du film. L'aspect matériel sera basé sur une carte DEO-nano-soc de Terasic ainsi qu'une (ou plusieurs) carte électronique d'interfaçage.

La partie logicielle réalisera le pilotage de la partie matérielle et devra rester synchronisée avec l'heure et la date courante à l'aide d'un serveur NTP (Network Time Protocol). La liaison avec le serveur se fera à minima à travers une liaison Ethernet. Si le temps le permet, l'étude de l'utilisation d'un modem 4G devra être réalisée afin d'offrir une alternative à au réseau filaire Ethernet. La partie logicielle doit être configurable à la

volée (pendant le fonctionnement). Une interface de type page web est à privilégier et doit être hébergée sur la carte DE0-Nano-Soc.

L'ensemble sera livré avec plusieurs documentations. Une première documentation complète doit présenter en détail la démarche de conception et spécification du système. Elle comprendra aussi l'ensemble des analyses fonctionnelles, et les détails d'architectures du système afin de permettre la maintenance et les modifications du système. L'ensemble des fonctionnalités seront détaillés à travers un manuel pour utilisateur avancé.

Un second manuel sera nécessaire pour le démonstrateur pendant les événements et permettra de mettre en place le système et le dépanner en cas de problème.

Enfin, une notice très réduite sera à fournir sous forme de fiche explicative à destination des utilisateurs finaux (visiteurs des salons).

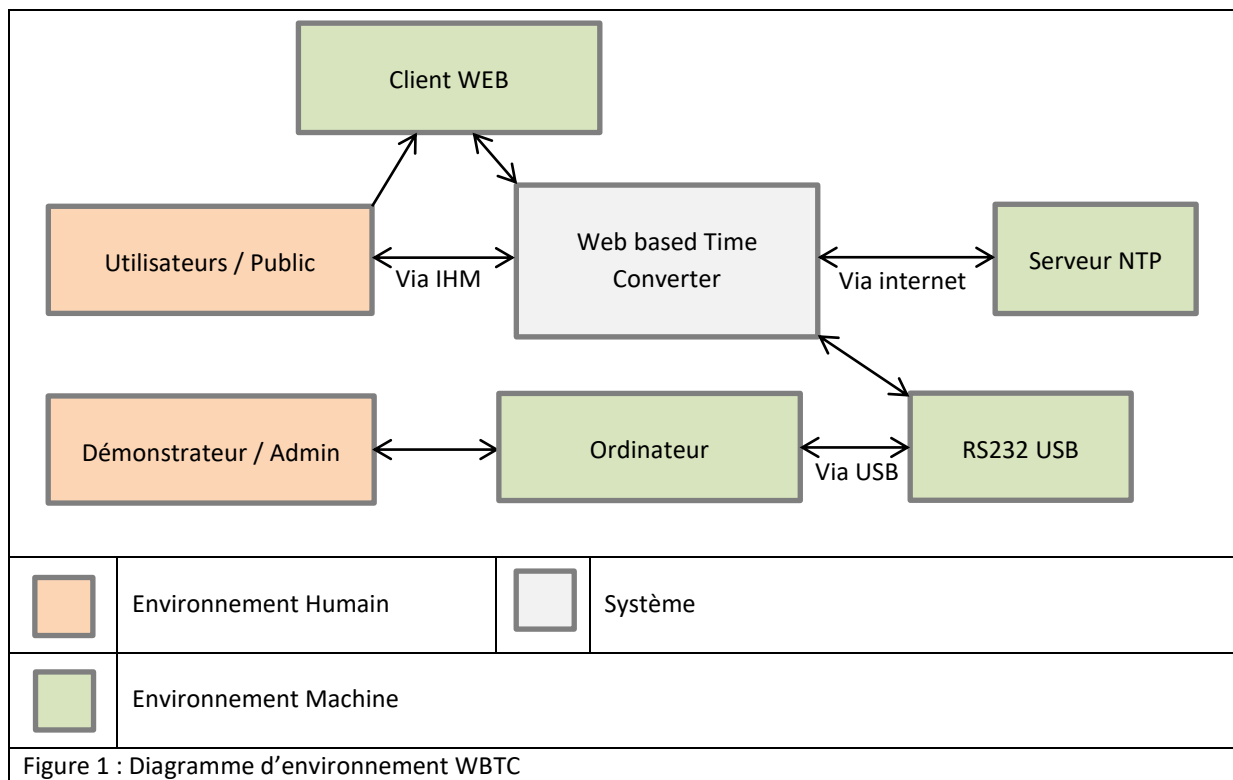
### 2.3. Bases méthodologiques

Les bases méthodologiques associées à la co-conception seront à mettre en place.

On parle notamment de l'utilisation des schémas blocs SysML, UML et System Analysis for Real Time.

## 3. Description générale

### 3.1. Environnement du projet



Le projet WBTC n'a pas d'existant. Cette année est effectivement une reprise du projet qui a été démarré l'an dernier. Il n'y a pas d'implication avec d'autres projets. Le système comporte en son centre le web based time converter. Il s'interfacera avec des serveurs NTP nécessaires à la synchronisation. De plus, les utilisateurs, c'est-à-dire le public des salons et le démonstrateur viendront interagir avec le système via un client WEB, une liaison RS232 USB et via l'IHM.

### 3.2. Caractéristiques des utilisateurs

On dénombre 3 utilisateurs du système :

1. Le **super-utilisateur** correspond au client *M.Rolland*. Il connaîtra le fonctionnement complet du système. Il sera à même, sur la base des documentations et de ses connaissances personnelles, de paramétrer, dépanner, modifier et, si besoin, mettre à jour le système.
2. Le **démonstrateur** utilisera le système en tant que support de présentation pendant les salons. Il sera capable de maintenir le système en fonctionnement le temps d'une ou plusieurs journées (mise en place et redémarrage par exemple).
3. Le **visiteur** n'aura accès qu'à un ensemble de fonctions de démonstrations détaillés au travers d'une fiche explicative.

### 3.3. Fonctionnalités et structure générale du système

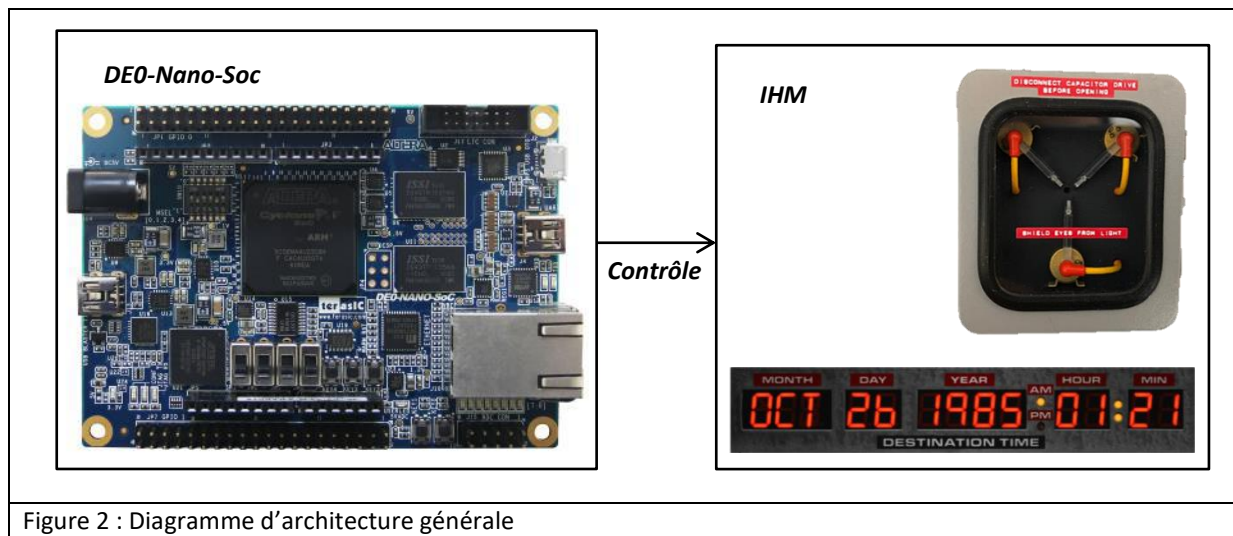


Figure 2 : Diagramme d'architecture générale

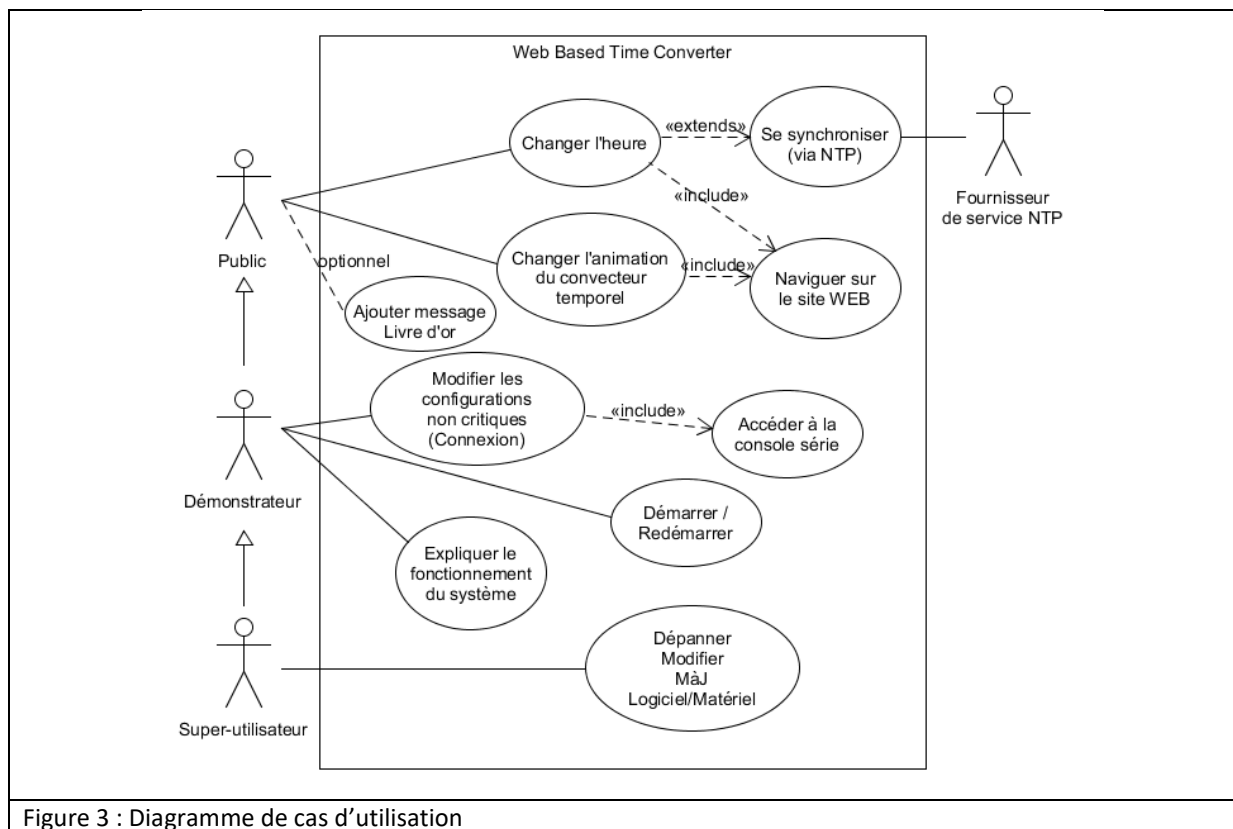


Figure 3 : Diagramme de cas d'utilisation

L'architecture ici présentée est en partie inspirée et reprise du travail de F. Rémy, ancien étudiant sur le projet. Le fournisseur de service NTP sera un acteur secondaire du système car il permettra de réaliser la synchronisation de l'horloge à l'aide d'un de ses serveurs NTP.

### **3.4. Contraintes de développement, d'exploitation et de maintenance**

#### **3.4.1. Contraintes de développement**

Les logiciels devront être conçus pour fonctionner sur une carte FPGA DE0-Nano-Soc composé d'un FPGA et d'un processeur ARM Cortex A9 associé à un système d'exploitation GNU/Linux.

Il y a deux à trois parties à retenir.

La première partie développement sera orientée pour l'OS Linux sur le Cortex A9 et n'impose aucun langage ni environnement de programmation. Ces programmes devront être cross-compilés ou compilés sur la cible.

La seconde partie concerne la synthèse matérielle implantée dans le FPGA qui devra donc être écrite dans un langage de type HDL (VHDL ou Verilog). Le logiciel Quartus sera à utiliser pour cette partie du développement. Le programme du FPGA sera chargé soit au démarrage du système Linux soit à l'aide d'un périphérique de chargement dédié de type Altera EPCS. Il faudra s'assurer d'utiliser la suite de développement de façon à produire les bons formats de fichiers.

Le développement matériel sera testé au maximum à travers des jeux de tests en simulation avant déploiement sur la cible.

La troisième partie éventuelle concerne le développement pour une cible microcontrôleur qui serait synthétisée dans le FPGA ou externe à celui-ci. Le développement passera ici par l'utilisation des langages de programmations C et C++ et devront être cross-compilé pour être chargés sur le microcontrôleur.

#### **3.4.2. Contraintes d'exploitation**

En ce qui concerne la sécurité, toutes les connexions entrantes et sortantes inutiles devront être désactivées. Ce blocage sera assuré par un pare-feu bloquant ces connexions.

Le système doit être robuste pour être opérationnel le plus longtemps possible et assurer une présentation prolongée des savoir-faire de l'école pendant les salons.

On peut dégager deux modes dégradés :

1. Mode IHM HS : l'interface homme machine matérielle est totalement ou partiellement inopérante mais le service WEB et la configuration de l'heure courante sont toujours opérationnels.
2. Mode IHM et Services HS : L'IHM ainsi que les services web sont inopérants. C'est un état grave dans lequel le système n'est plus fonctionnel du tout.

Pour ces cas, il faudra prévoir un mode de récupération et dépannage du système permettant de rétablir la fonctionnalité. L'idéal serait qu'un redémarrage permette de rétablir l'état opérationnel du système. Un système de log sera disponible pour permettre une analyse des défaillances sur place ou à posteriori.

Il faudra aussi qu'un redémarrage système place le système dans un état indépendant de l'état dans lequel il était précédemment.

## **4. Description des interfaces externes du logiciel**

### **4.1. Interfaces matériel/logiciel**

L'interface logiciel-matériel sera réalisée par le FPGA et l'électronique associée. Le pilotage se fera via les routines et couches de lecture et écriture entre le FPGA et le Cortex A9. L'IHM étant contrôlée par le FPGA, le



pilotage passera forcément par celui-ci. Il y a quelques périphériques tels que la connectivité Ethernet et USB qui pourront passer nativement par le cœur A9 et permettront d'accéder à Internet.

Concernant le type d'accès aux services Web et NTP pour la synchronisation de l'heure, une connexion sous forme de socket TCP/IP sera à envisager à défaut de gestion native par l'OS.

La connexion entre la carte DE0-nano-Soc passera par une liaison RS232 sur USB pour une connexion native à la carte. Cette connexion permettra notamment d'avoir accès aux informations de démarrage du système en temps réel. Etant incluse à la carte, cette fonctionnalité n'aura pas besoin d'attention particulière. Sinon, à l'exploitation, il sera possible d'utiliser un tunnel SSH sécurisé pour administrer le système par le réseau Ethernet. De la même façon, les services Web de configuration seront accessibles par le biais de cette connectivité Ethernet. Ces accès seront détaillés plus bas dans les interfaces logiciel-logiciel.

## 4.2. Interfaces homme/machine

L'interface homme machine sera composée de 3 éléments :

1. L'IHM non interactive sera représenté par le système d'affichage d'heure et date courante ainsi que du convecteur temporel. Ces éléments sont des composants visuels lumineux et aucune interaction telle que le toucher n'est attendu sur ces éléments. En termes d'intelligence sur ces parties, on évitera au maximum l'utilisation d'intelligence que l'on préférera intégrer dans le FPGA. L'intelligence intégrée pourra se réduire à l'utilisation de composants électroniques dit intelligent par l'utilisation d'interface de type SPI ou I2C. Le convecteur temporel imposé est composé de LEDs intelligentes adressables WS2812b en est un autre exemple. Cet élément s'inspirera fortement du film *Retour Vers Le Futur* et représente la partie imposée de l'IHM.
2. L'IHM interactive sera composée d'éléments possiblement placés sur la partie libre de l'IHM ainsi que d'une interface de configuration sur basée idéalement sur un site Web accessible depuis un navigateur Web.

## 4.3. Interfaces logiciel/logiciel

### 4.3.1. Serveur NTP

Un premier élément d'interface logiciel-logiciel est associé à la connexion à un serveur NTP pour la synchronisation du temps du système. Le principe est d'utiliser des technologies standard des systèmes GNU/Linux pour réaliser cette interface. On pense notamment au daemon ntpd capable de réaliser cette tâche en natif. Si toutefois cette fonction n'est pas implémentée dans l'OS, on utilisera une connexion à travers des sockets TCP/IP et les bibliothèques associées.

### 4.3.2. Client Web

Les clients WEB seront externes au système et viendront se connecter au serveur WEB déployé sur la carte DE0-Nano. Cette connexion permet la configuration du système avec notamment la saisie de l'heure et date courante, le choix du serveur NTP sur lequel se synchroniser ainsi que la configuration lumineuse du convecteur temporel. L'instance WEB sera accessible sans installation sur le poste dédié aux visiteurs. On pourra même imaginer une utilisation sur via le navigateur web mobile d'un smartphone. Il faudra cependant s'assurer de mettre en place un système de gestion des connexions simultanées. Ceci évitera que les connexions ne soient modifiées en même temps par deux utilisateurs. A défaut de ne restreindre les accès qu'à un client à la fois, on devra gérer les modifications concurrentes. Une connexion spécifique allouée à l'administrateur sera quant à elle toujours disponible pour permettre le dépannage ou la réinitialisation du système. On peut aussi envisager d'utiliser le client WEB comme un moyen d'accès à un livre d'or dans lequel le visiteur pourrait laisser un message.

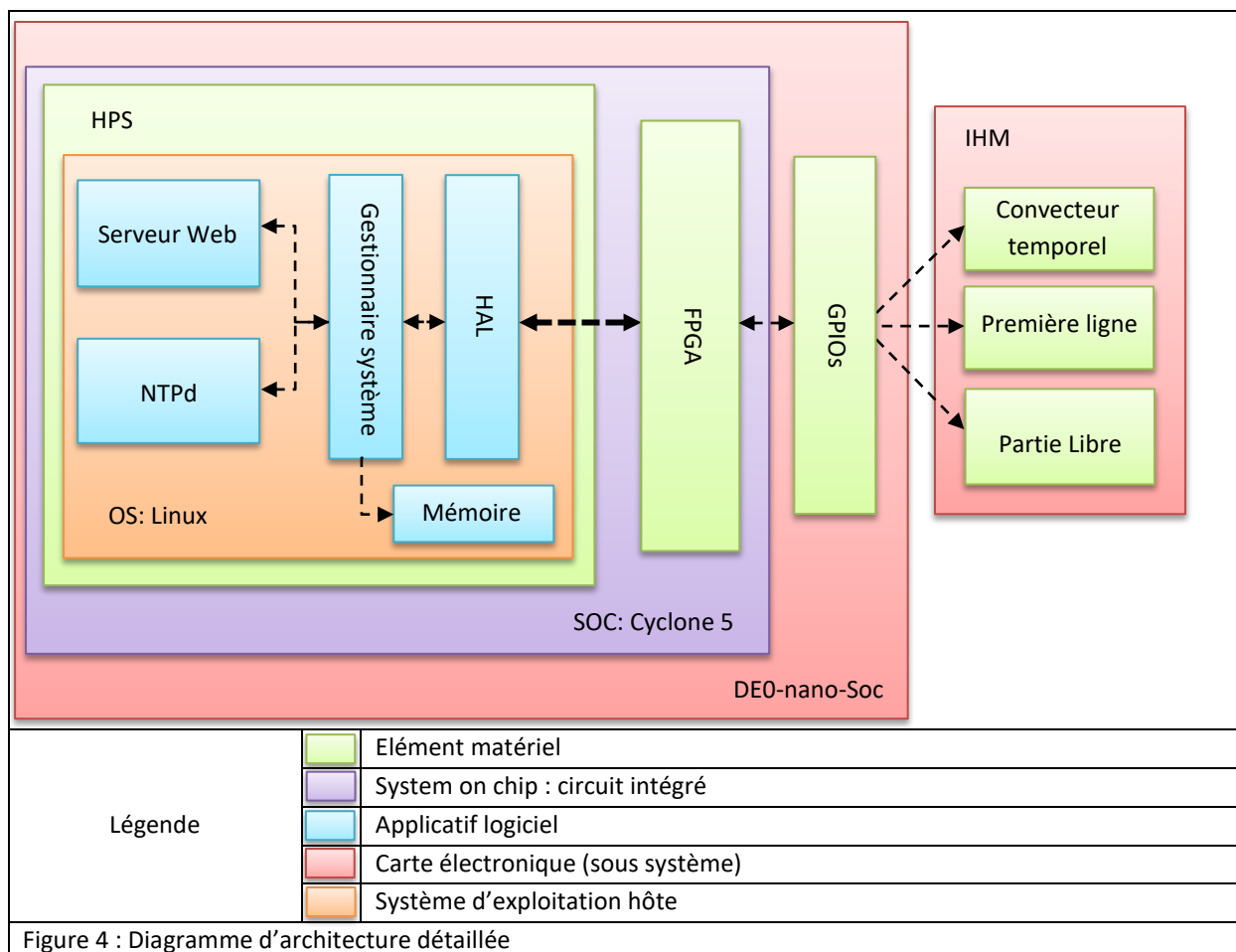
#### 4.3.3. Gestionnaire de configuration système

Le gestionnaire de configuration système sera une couche logicielle intégrée au système et permettra de gérer l'état du système. Il gardera en mémoire les configurations des services Web, de l'état des périphériques de l'IHM et permettra d'agir sur le matériel à travers la couche d'abstraction matérielle. De la même façon, le serveur web ne sera pas à même d'agir directement sur la couche d'abstraction matérielle mais demandera des changements de configuration du système (i.e. changer le serveur NTP de synchronisation). Cette couche de gestion interne sera pilotable par différents biais de communication. On peut imaginer qu'une interface à ligne de commande soit disponible pour gérer l'état du système ainsi qu'une interface sur socket UNIX ou TCP/IP accessible depuis d'autres processus du système pour les mêmes actions.

#### 4.3.4. Couche d'abstraction matérielle (HAL)

La couche d'abstraction matérielle (HAL) est la dernière l'interface entre les programmes haut niveau et l'interface logiciel-matérielle. C'est par cette couche accessible par ligne de commande ou communication interprocessus que le transfert du pilotage sera effectué. Ainsi, cette couche d'abstraction permettra de tester le système sans avoir de matériel associé tout en s'assurant que les messages transitent bien et demandent les bonnes actions sur l'IHM. On assure aussi la modularité dans le système par ce bien.

### 5. Architecture générale du système



La carte DE0-Nano-Soc sera le point central d'intelligence du système. Le point d'entrée pour le pilotage et la modification des paramètres sera le serveur WEB qui agira sur un gestionnaire système. Le gestionnaire

système connaîtra en tout temps la configuration du système et s'assurera d'appliquer les changements d'état à l'IHM à travers la HAL. Enfin, l'ensemble de l'IHM sera piloté par le FPGA. La couche d'abstraction matérielle devra mettre en œuvre une communication entre le HPS et le FPGA depuis le système Linux. C'est donc une interface logicielle-matérielle critique et qui passe par les couches de l'OS Linux, puis à travers les ponts HPS vers FPGA et réciproquement.

## 6. Description des fonctionnalités

<b>Nom</b>	Configurer la synchronisation horaire
<b>Rôle</b>	Cette fonction permet à l'utilisateur de choisir le fuseau horaire sur lequel le système doit se synchroniser ainsi que le serveur de synchronisation.
<b>Priorité</b>	Prioritaire
<b>Entrées</b>	Liste des fuseaux horaires disponibles pour la synchronisation. Liste des serveurs NTP possibles pour la synchronisation.
<b>Sorties</b>	Modification de la configuration du fuseau horaire et synchronisation sur un des serveurs dès que possible. L'IHM sera mise à jour en temps réel à la synchronisation.
<b>Fonctionnement prévu</b>	Le programme va interroger une base de données contenant la liste des serveurs actifs. Les serveurs NTP donnant l'heure UTC, il ne sera pas nécessaire de trier les serveurs par fuseau horaire. L'utilisateur ne renseignera qu'un fuseau horaire et le système sera mis à jours en conséquence. L'état de synchronisation pourra être visible. L'utilisateur n'aura pas la vision sur la liste des serveurs NTP disponible à la synchronisation, mais l'administrateur pourra renseigner explicitement les serveurs à utiliser.
<b>Erreur</b>	Si aucune connexion n'est disponible avec un des serveurs NTP, l'interface WEB offrira la possibilité de changer l'heure du système. Il pourra être possible de choisir de resynchroniser l'heure lorsqu'un des serveurs sera disponible.

<b>Nom</b>	Piloter la première ligne d'affichage
<b>Rôle</b>	La première ligne d'affichage affichera l'heure à l'aide d'afficheurs 7 et 16 segments. L'objectif de cette fonctionnalité est de pouvoir piloter ce matériel à l'aide de l'OS Linux. On doit donc pouvoir renseigner les chiffres et lettres à afficher depuis Linux et voir les changements répercutés sur l'IHM en temps réel. Si possible, on construira un programme qui en se base sur l'heure et date de l'OS pour l'afficher sur les afficheurs. Elle concernera uniquement le système et l'administrateur.
<b>Priorité</b>	Prioritaire
<b>Entrées</b>	Valeurs des afficheurs 16 et 7 segments : Mois, Jours, Année, Heure, Minute, AM/PM, Indicateur de secondes.
<b>Sorties</b>	Mise à jour des afficheurs 16 et 7 segments de l'IHM.
<b>Fonctionnement prévu</b>	Les informations à afficher sur l'IHM (date/heure) seront passées à travers un appel à une commande ou une liaison IPC. Le programme réalisera ensuite la communication avec le FPGA pour mettre à jours les afficheurs 7 et 16 segments de l'IHM. C'est un pilote matériel qui affichera les données renseignés sur l'IHM. Cette fonction doit pouvoir mettre à jour l'IHM en moins de 300 ms permettant d'atteindre une fréquence de rafraichissement de 3Hz minimum.
<b>Erreur</b>	Le pilotage des éléments ayant de fortes chances d'être réalisés en boucle ouverte, il n'y aura que peu d'erreurs à prendre en compte. La gestion sera minimale.

<b>Nom</b>	Piloter le convecteur temporel
------------	--------------------------------

<b>Rôle</b>	Cette fonction sera en charge des motifs à afficher sur le convecteur temporel. Elle concernera uniquement le système et l'administrateur.
<b>Priorité</b>	Prioritaire
<b>Entrées</b>	Séquence de motifs à afficher sur le convecteur temporel
<b>Sorties</b>	Affichage des motifs un par un sur le convecteur temporel
<b>Fonctionnement prévu</b>	Cette fonction se chargera d'afficher les motifs demandés sur le convecteur temporel. Une séquence pourra donc être générée du côté Linux et transmise via le FPGA aux LEDs du convecteur temporel. Cette fonction correspond donc à un pilote matériel pour le convecteur temporel. Cette fonction doit pouvoir mettre à jours le convecteur temporel à une fréquence de 60Hz minimum pour garantir le fonctionnement avec des animations rapides.
<b>Erreur</b>	Le pilotage des éléments ayant de fortes chances d'être réalisés en boucle ouverte, il n'y aura que peu d'erreurs à prendre en compte. La gestion sera minimale.

<b>Nom</b>	Modifier les animations du convecteur temporel
<b>Rôle</b>	Permettre à l'utilisateur de choisir un mode d'éclairage du convecteur temporel parmi une liste. L'utilisateur pourra choisir la palette de couleurs à utiliser.
<b>Priorité</b>	Secondaire
<b>Entrées</b>	Liste des animations associées à chaque séquence d'éclairage du convecteur temporel
<b>Sorties</b>	Une animation jouée sur le convecteur temporel
<b>Fonctionnement prévu</b>	Une séquence d'éclairage du convecteur temporel correspond à un algorithme ou une séquence d'images à afficher sur les LED du convecteur temporel. L'utilisateur accèdera à une liste de choix d'animations associées à certains paramètres optionnels (vitesse, couleur(s), intensité lumineuse). Lorsque l'utilisateur aura choisi sa séquence, le comportement du convecteur temporel sera mis à jour et affichera la nouvelle animation. L'utilisateur pourra aussi choisir d'éteindre le convecteur. L'administrateur pourra ajouter, supprimer et modifier les animations à rendre disponible au public.
<b>Erreur</b>	Les valeurs de paramètres pour les animations devront être valides pour chaque animation. L'envoi de valeurs invalide ne devra pas mettre en péril la stabilité du système.

<b>Nom</b>	Modifier le style d'affichage
<b>Rôle</b>	Cette fonction permet de changer le style de la ligne d'afficheur. (24h ou 12h + AM/PM, style de clignotement des points heure/minutes, et langue à utiliser pour les mois)
<b>Priorité</b>	Optionnelle
<b>Entrées</b>	Liste de configuration de l'afficheur : <ol style="list-style-type: none"> <li>1. 24h – 12h</li> <li>2. Style de clignotement des points : On , Off, fréquence, compteur</li> <li>3. Langue</li> </ol>
<b>Sorties</b>	Mise à jour du style de la première ligne d'afficheur en temps réel avec le nouveau style.
<b>Fonctionnement prévu</b>	L'utilisateur pourra choisir le mode d'affichage de l'heure sur la ligne d'affichage à savoir au format 24h (FR) ou 12h AM/PM (EN). Il sera aussi possible de configurer le clignotement des points entre heure et minutes. La langue sera paramétrable avec à minima les langues Français et Anglais. La langue aura un impact sur l'affichage alpha numérique du mois.
<b>Erreur</b>	Si une langue sélectionnée n'est pas implémentée, le système passera sur la langue anglaise.

<b>Nom</b>	Ajouter un message dans le livre d'or
------------	---------------------------------------

<b>Rôle</b>	Récolter les avis et messages du public pendant les forums.
<b>Priorité</b>	Optionnelle
<b>Entrées</b>	Message de l'utilisateur
<b>Sorties</b>	Stockage du message en base de données
<b>Fonctionnement prévu</b>	L'utilisateur pourra à travers le site WEB du système accéder à une interface de saisie d'un message dans le livre d'or. Il pourra écrire un commentaire qui sera stocké dans une base de données à dimensionner. Une prévisualisation des messages laissés par les autres utilisateurs sera disponible (par exemple les 5 derniers messages) pour inspirer l'utilisateur actuel à saisir un message. L'administrateur pourra consulter, modifier, exporter et supprimer les messages du livre d'or.
<b>Erreur</b>	Si la base de données est pleine, la fonctionnalité de saisie du message sera bloquée. La prévisualisation sera quant à elle toujours disponible dans ce cas de figure. Si la base de données est vide, il n'y aura pas de message dans la prévisualisation.

<b>Nom</b>	Modifier les configurations non-critiques
<b>Rôle</b>	Permet au démonstrateur et au super-utilisateur de modifier les informations de connexion qui peuvent varier d'un salon à l'autre. D'autres configurations pourront être appliquées pour s'adapter aux différences de chaque salon.
<b>Priorité</b>	Secondaire
<b>Entrées</b>	Interface de configuration des paramètres initialisée à la configuration actuelle du système.
<b>Sorties</b>	Nouvelle configuration du système appliquée à la suite de validation par l'utilisateur. (Pas de changement à la volée)
<b>Fonctionnement prévu</b>	La configuration passera par le port série RS232 USB et un terminal série sur un PC. L'interface de configuration sera lancée à l'exécution du script de configuration. L'utilisateur renseignera alors ses paramètres et pourra les appliquer par validation de ces derniers. La validation entraînera une modification automatique des paramètres sur le système.
<b>Erreur</b>	L'utilitaire de configuration n'autorisera pas l'utilisateur à rentrer des valeurs absurdes où pouvant laisser le système dans un état indéterminé. Les erreurs de saisie seront donc écartées. Les erreurs ne seront pas générées par l'utilitaire de saisie mais éventuellement pas les composants externes qui seront affectés par les changements de configuration.

## 7. Conditions de fonctionnement

### 7.1. Performances

Le système ayant vocation à être utilisé sur des salons et aux portes ouvertes, on attend qu'il soit fonctionnel en continu sur la durée de l'évènement (1 week-end complet minimum). **Le temps de réponse** doit être **inférieur à la seconde** pour les **changements de configuration du système**. Un temps supérieur pourrait être perçu comme un dysfonctionnement. **Les animations du convecteur temporel** avec un **taux de rafraichissement allant jusqu'à 60Hz** devront fonctionner sans problème de performance visible (lag ou jitter bien inférieur à la fréquence de rafraichissement). **Le système d'affichage d'heure et date** pourra être mis à jour avec une **fréquence de 5Hz minimum** pour assurer les animations des éléments de l'afficheur.

Par rapport à l'environnement du système, une synchronisation aux serveurs NTP est à prévoir aussi souvent que possible. **A chaque reprise de connectivité internet**, le système devra se resynchroniser **dès que les requêtes NTP seront en mesure de passer** sur le réseau. **Les changements de fuseau horaire** n'agissant pas sur les serveurs NTP mais sur la configuration locale du système, on estime qu'un **temps de réponse inférieur à la seconde** sera attendu pour cette modification.

Les changements de configuration pourront être appliqués plusieurs centaines de fois par jours sans dégradation de performance ou perte de stabilité. On estime que chaque visiteur souhaitera changer la configuration du système. A raison de 25 personnes par heure, cela donnera environ 100 à 150 personnes par jour.

## 7.2. Capacités

Le système étant une cible embarquée à performances limitées, on essayera de limiter les connexions simultanées pour éviter les cas de plantage. Par ailleurs, la connexion au serveur Web sera toujours possible pour un administrateur. L'usage simultané par plusieurs visiteurs pourra être rendu possible mais limité à 10 connexions maximales. La concurrence des modifications sera à prendre en compte pour éviter les corruptions de données. Deux utilisateurs pourront changer les paramètres en même temps mais c'est le serveur WEB qui départagera laquelle des deux configurations sera appliquée. Quoi qu'il puisse arriver en termes de modification concurrentes, le système devra au mieux éviter le plantage. Si possible, si un utilisateur change un paramètre, les autres utilisateurs verront le nouveau paramètre à jour sur leur page web.

Si le livre d'or est implémenté, on fera en sorte de stocker la base de données sur le système et non en externe pour pallier aux risques de perte de connectivité. La base de donnée ainsi que sa taille devront être adaptés à contenir au moins 500 enregistrements (l'équivalent de 3-4 jours de salons) et plus dans les capacités offertes par la carte DE0-Nano-Soc.

## 7.3. Modes de fonctionnement (optionnel)

Après une mise sous tension, la carte DE0-Nano-Soc va lancer l'OS Linux présent sur la carte micro-SD. S'il n'y a pas de carte SD, la carte DE0-Nano-Soc restera dans un état non initialisé. Lorsque l'OS linux démarre, une procédure va programmer la configuration du FPGA. Une fois démarré, le système lancera le gestionnaire système, les services NTP, et les programmes utiles. Au démarrage du gestionnaire du système, il appliquera la configuration sur l'IHM.

En cas de rupture d'alimentation, s'il n'est pas possible de disposer du temps nécessaire pour sauver le système, on fera en sorte d'avoir précédemment sauvé la configuration du système par des sauvegardes régulières. Si la configuration est corrompue, une tentative de retour à la configuration d'usine sera effectuée. Le système utilisant une carte SD pour le système Linux, on essayera d'utiliser une carte industrielle plus résistante aux coupures de courant. Dans tous les cas, une coupure de courant inattendue sera à éviter autant que possible au risque de détruire la carte SD. On essayera de mettre en place un bouton sur le site web facile d'accès pour l'administrateur pour éteindre rapidement le système sans passer par la console RS232. Un bouton physique serait aussi intéressant à déployer. On pourrait ainsi rapidement éteindre proprement le système avant de débrancher l'alimentation. Le FPGA étant réinitialisé après une coupure de courant, il ne sera pas nécessaire de l'éteindre proprement. Un témoin de mise hors tension sera à mettre en place pour assurer que le système est éteint.

Le système pourra fonctionner dans deux modes dégradés. Le premier mode dégradé dans lequel un des éléments de l'IHM n'est plus opérationnel permettra quand même de réaliser la démonstration et l'interaction utilisateur pourra continuer avec moins de fonctions. L'utilisateur pourra toujours laisser un message dans le livre d'or mais ne pourra pas par exemple agir sur le convecteur temporel. Le deuxième mode dégradé est celui dans lequel l'ensemble des éléments d'IHM sont hors services. Ceci implique qu'une interaction utilisateur ne sera possible, un dépannage au plus vite sera nécessaire pour remettre le système en marche. L'interaction utilisateur étant l'intérêt principal du système, ce deuxième mode dégradé sera à éviter autant que possible.

## 7.4. Contrôlabilité

On s'attachera à garder les informations de connexions sous forme de log afin de déterminer les responsabilités en cas de problème légal. Il sera nécessaire de garder des logs de fonctionnement afin de pouvoir déterminer l'état du système à tout moment et surtout en cas de plantage. S'il est possible d'enregistrer la chaîne d'évènements qui a mené à la défaillance, on le fera. Ces informations seront très utiles pour résoudre les problèmes.

### 7.5. Sécurité

En termes de sécurité, il y a différents niveaux d'accès au système. Le public (les visiteurs de salons) n'aura accès qu'au service WEB de pilotage de l'IHM ne n'aura de fait que le droit d'interagir avec elle par les simples outils mis à sa disposition. Il ne devra pas y avoir moyen pour cet utilisateur de modifier d'autres configurations du système ou d'accéder au système de fichier.

Seul le démonstrateur et le super-utilisateur seront à même d'accéder au système par le biais d'un terminal série et de la liaison RS232-USB ou une connexion SSH. La première configuration du système nécessitera le port RS232 mais une fois le réseau configuré et l'interface connectée, une connexion SSH sécurisée sera envisageable. Seuls le démonstrateur et le super-utilisateur seront détenteurs des identifiants de connexion. On favorisera tout de même l'accès physique plus difficile d'accès pour des personnes non autorisées (surtout s'il faut démonter un boîtier pour accéder à la carte). Le démonstrateur n'aura toutefois pas tous les droits sur le système de fichier. Pour la suppression de fichiers ou les modifications de paramètres critiques, il sera nécessaire de passer par le super-utilisateur.

### 7.6. Intégrité

Les pertes de courant pouvant être fatales pour la carte micro-SD, support de stockage des bases de données et du système, on s'attachera à livrer une image complète du système associée à une documentation de restauration en cas de plantage. Il suffira alors d'acheter une nouvelle carte micro-sd de même capacité que la précédente et restaurer l'image SD sur celle-ci.

Le risque de pertes de données et fuites pourrait arriver avec un piratage de la base de données où le vol d'au moins la carte micro-SD du système et au pire le système entier. On fera en sorte que le second problème n'arrive pas en laissant le matériel sous surveillance et en rendant l'accès à la carte micro-SD suffisamment difficile. De même, la base de données ne devra pas être ouverte au public et de fait, les connexions extérieures à la carte elle-même seront refusées. On évitera au minimum les possibilités de fuites.

Par ailleurs, la base de données n'étant sujette qu'à stocker des données de commentaires les plus anonymes possible, on estime que le vol des informations n'aura que peu de conséquences.

## PLAN DE DEVELOPPEMENT

### 8. Découpage du projet en tâches

<b>Nom</b>	Définition / Lancement projet
<b>Identifiant</b>	1
<b>Description de la tâche</b>	Reprendre le projet, récupérer ce qui a été fait l'an dernier, lire les documents et comprendre le travail accompli. Récupération du matériel. Expression des besoins client.
<b>Cycle de vie</b>	Reprise projet, Réunion de lancement
<b>Livrables</b>	Notes de réunion de lancement
<b>Estimation de charge</b>	4 j/h
<b>Contraintes temporelles</b>	Les 2 premières semaines après le lancement du projet.

<b>Nom</b>	Rédaction du cahier de spécifications
<b>Identifiant</b>	2
<b>Description de la tâche</b>	Rédaction du cahier de spécifications du projet et les détails des tenants et aboutissants du projet WBTC.
<b>Cycle de vie</b>	Après reprise projet, Rédaction, Validation en fin de rédaction.
<b>Livrables</b>	Cahier de spécifications rédigé au format PDF
<b>Estimation de charge</b>	6 j/h
<b>Contraintes temporelles</b>	Tâche à réaliser avant le 15/11/19

<b>Nom</b>	Réalisation IHM : Date et Heure (Time Circuit Display)
<b>Identifiant</b>	3
<b>Description de la tâche</b>	<p>Reprise des éléments d'analyse de l'an dernier, dégager les points importants et objectifs avec le client pour construire l'IHM d'affichage d'heure et date. Le Time Circuit Display (TCD) est à base de 16 et 7 segments. Avec les attendus de l'IHM énoncés on analyse et compare :</p> <ol style="list-style-type: none"> <li>1. Comparer les méthodes de pilotage d'afficheurs 7 segments (pilotage en puissance, interfaçage, coût) : recherches documentaires et veille technologique</li> <li>2. Choisir une ou plusieurs méthodes candidates pour le pilotage</li> <li>3. Analyser la méthode appliquée au TCD avec un schéma fonctionnel</li> <li>4. Réaliser une analyse structurelle (avec choix de composants)</li> <li>5. Analyser la méthode de pilotage logicielle : interfaçage, librairie logicielle</li> <li>6. Commande de matériel</li> <li>7. Monter et tester sur platine d'essai avec codage logiciel sur carte de développement (type Arduino).</li> <li>8. Routage d'un PCB</li> <li>9. Validation par M.Rolland</li> <li>10. Fabrication du PCB</li> <li>11. Montage et test du PCB (idem que breadboard)</li> </ol>
<b>Cycle de vie</b>	Veille technologique, Analyse, Choix de composants, Fabrication et Tests
<b>Livrables</b>	PCB complet du Time Circuit Display ressemblant à celui du film Retour vers le futur



<b>Estimation de charge</b>	5j/h
<b>Contraintes temporelles</b>	Il faut commander les composants à minima pour expérimentation avant le 15/11/19 qui est la clôture des budgets de l'école.

<b>Nom</b>	Réalisation IHM : Flux Capacitor (Convecteur temporel)
<b>Identifiant</b>	4
<b>Description de la tâche</b>	<p>Récupération du convecteur temporel fabriqué par M.Rolland (à base de WS2812) et le mettre en œuvre avec des animations similaires à celle du film.</p> <ol style="list-style-type: none"> <li>1. Analyse des animations du film : vitesse, couleur, fréquence, forme</li> <li>2. Réalisation d'algorithmes d'animations pour la commande des couleurs des LED du convecteur.</li> <li>3. Codage des algorithmes sur une carte de développement (type Arduino).</li> <li>4. Valider les animations avec M.Rolland</li> <li>5. Réaliser un driver de LED WS2812 matériel (HDL) pour la carte DE0-Nano-Soc</li> <li>6. Porter les animations sur la carte DE0-Nano-Soc</li> </ol>
<b>Cycle de vie</b>	Veille technologique, Analyse logicielle, Programmation C, Programmation HDL, Test
<b>Livrables</b>	Un élément lumineux à base de LED WS2812 ayant des animations similaires à celui de la De Lorean du film.
<b>Estimation de charge</b>	6j/h
<b>Contraintes temporelles</b>	-

<b>Nom</b>	Prise en main carte DE0-Nano-Soc
<b>Identifiant</b>	5
<b>Description de la tâche</b>	<p>Prendre en main la carte DE0-Nano-Soc, c'est-à-dire réaliser les premiers Hello World avec le FPGA, le HPS, installer l'OS Linux. Pour cela, il faut :</p> <ol style="list-style-type: none"> <li>1. Rechercher la documentation et des tutoriels</li> <li>2. Installer l'environnement de développement</li> <li>3. Récupérer l'ensemble des fichiers liés à la carte DE0-Nano pour le développement</li> <li>4. Réaliser la connexion JTAG entre la carte et le PC de développement</li> <li>5. Ecrire un premier code Verilog, l'implanter dans le FPGA et vérifier le fonctionnement</li> <li>6. Installer une distribution Linux sur la carte micro-SD</li> <li>7. Démarrer le système Linux et le prendre en main</li> <li>8. Cross-compiler un Hello World sur le PC de développement</li> <li>9. Le déployer sur la carte DE0 via SSH ou micro-SD</li> <li>10. Tester le Hello World</li> </ol>
<b>Cycle de vie</b>	Recherche documentaires, programmation, installation, tests
<b>Livrables</b>	Procédure d'installation de l'environnement de développement, et premier codes de tests.
<b>Estimation de charge</b>	2 j/h
<b>Contraintes temporelles</b>	-

<b>Nom</b>	Réalisation IHM : Pilotage IHM via FPGA
<b>Identifiant</b>	6
<b>Description de la tâche</b>	<p>L'objectif de cette tâche est de pouvoir réaliser le pilotage des éléments d'IHM à travers le FPGA. Cette partie comprend plusieurs éléments :</p> <ol style="list-style-type: none"> <li>1. Analyse fonctionnelle</li> <li>2. Analyse des interfaces et signaux de commandes des éléments d'IHM</li> <li>3. Codage de modules de pilotage et librairies bas niveau</li> <li>4. Validation en simulation pour les parties FPGA et réel oscilloscope ou sur matériel si simulation impossible</li> <li>5. Analyse de la communication HPS &lt;-&gt; FPGA</li> <li>6. Hello World HPS &lt;-&gt; FPGA</li> <li>7. Analyse de la HAL</li> <li>8. Conception protocole de commandes HPS &lt;-&gt; FPGA</li> <li>9. Codage de la HAL : CLI dans un premier temps et IPC ensuite</li> <li>10. Tests de la HAL en pilotant le matériel</li> </ol>
<b>Cycle de vie</b>	Prédécesseurs : 3,4,5 , analyse, conception, codage, tests
<b>Livrables</b>	Documentation d'utilisation de la HAL et des modules d'IHM.
<b>Estimation de charge</b>	5 j/h
<b>Contraintes temporelles</b>	-

<b>Nom</b>	Réalisation Système : Serveur Web et Gestionnaire système
<b>Identifiant</b>	7
<b>Description de la tâche</b>	<p>L'objectif de cette tâche est de pouvoir réaliser créer le site WEB de pilotage du système associé au gestionnaire d'état.</p> <ol style="list-style-type: none"> <li>1. Analyse fonctionnelle : Front end et Back end, Base de données</li> <li>2. Analyse du gestionnaire système</li> <li>3. Conception : Front Back BDD gestionnaire système</li> <li>4. Maquette d'interface frontend du serveur</li> <li>5. Validation des maquettes par M.Rolland</li> <li>6. Recherche de framework de création de site WEB compatible ARM 32bits embarqué</li> <li>7. Hello World framework WEB</li> <li>8. Codage du front end</li> <li>9. Codage du back end</li> <li>10. Codage du gestionnaire système</li> <li>11. Cross compilation pour DE0-Nano</li> <li>12. Intégration sur la carte DE0</li> </ol>
<b>Cycle de vie</b>	Prédécesseurs : 5
<b>Livrables</b>	Logiciels déployés sur la carte DE0 nano et documentation d'utilisation du service WEB.
<b>Estimation de charge</b>	8 j/h
<b>Contraintes temporelles</b>	-

<b>Nom</b>	Rédaction cahier d'analyse
<b>Identifiant</b>	8
<b>Description de la tâche</b>	L'objectif de cette tâche est de réaliser un cahier d'analyse qui retrace les

	réflexions menées pendant le projet pour faciliter une éventuelle reprise de celui-ci par un futur étudiant.
<b>Cycle de vie</b>	-
<b>Livrables</b>	Cahier d'analyse au format PDF
<b>Estimation de charge</b>	8 j/h
<b>Contraintes temporelles</b>	1 semaine avant la soutenance

<b>Nom</b>	Rédaction rapport de projet
<b>Identifiant</b>	9
<b>Description de la tâche</b>	L'objectif de cette tâche est de réaliser un cahier d'analyse qui retrace les réflexions menées pendant le projet pour faciliter une éventuelle reprise de celui-ci par un futur étudiant.
<b>Cycle de vie</b>	-
<b>Livrables</b>	Rapport de projet au format PDF
<b>Estimation de charge</b>	8 j/h
<b>Contraintes temporelles</b>	1 semaine avant la soutenance

<b>Nom</b>	Suivi et gestion projet
<b>Identifiant</b>	10
<b>Description de la tâche</b>	L'objectif de cette tâche est de faire apparaître le temps nécessaire à la gestion de projet, c'est-à-dire le suivi des avancements, la communication avec le client et la mise en œuvre des techniques de gestion de projet de l'ingénieur.
<b>Cycle de vie</b>	Suivi tout au long du projet
<b>Livrables</b>	Rapport de projet au format PDF
<b>Estimation de charge</b>	5 j/h
<b>Contraintes temporelles</b>	1 rapport hebdomadaire à l'encadrant

## 9. Planning

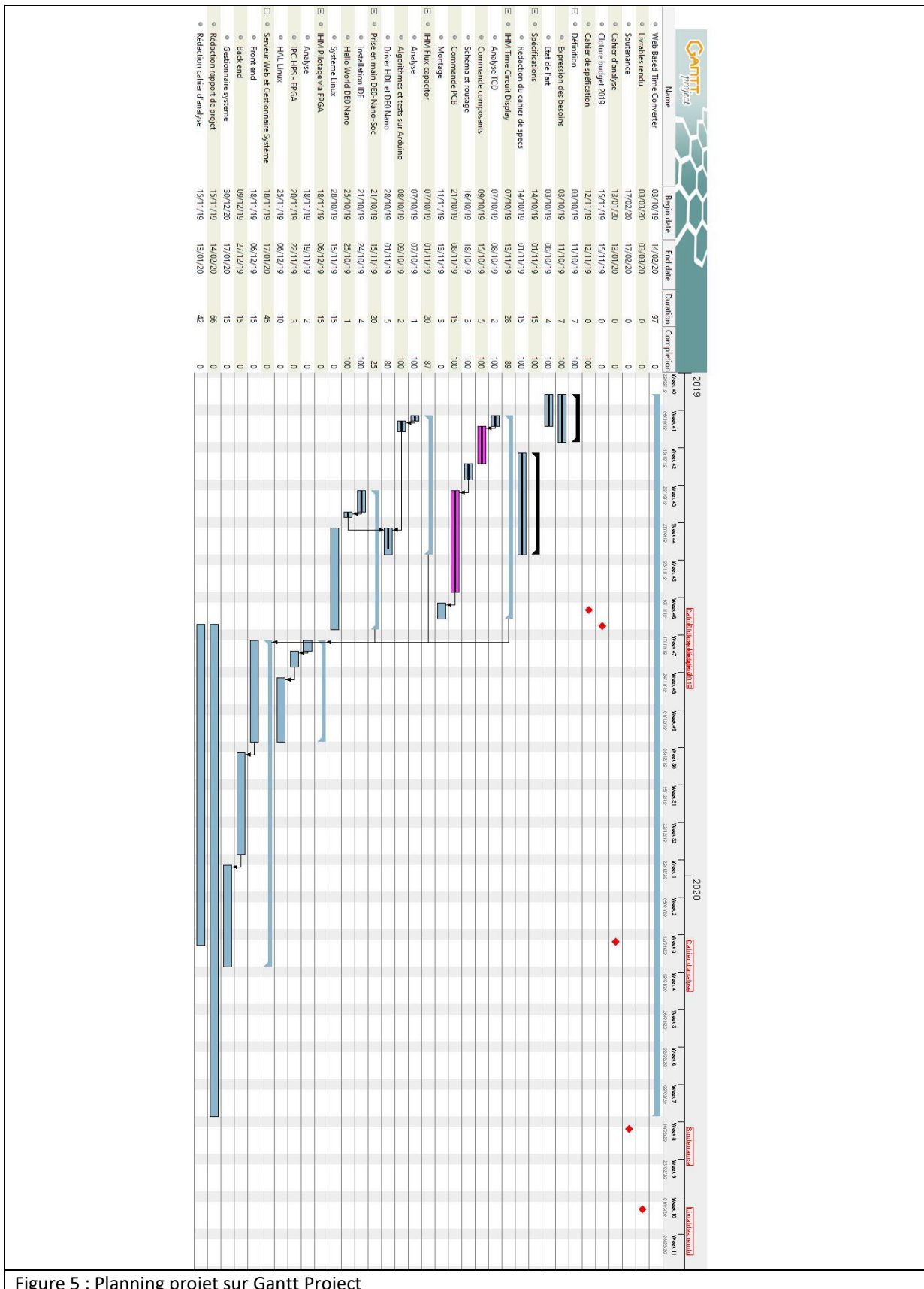


Figure 5 : Planning projet sur Gantt Project

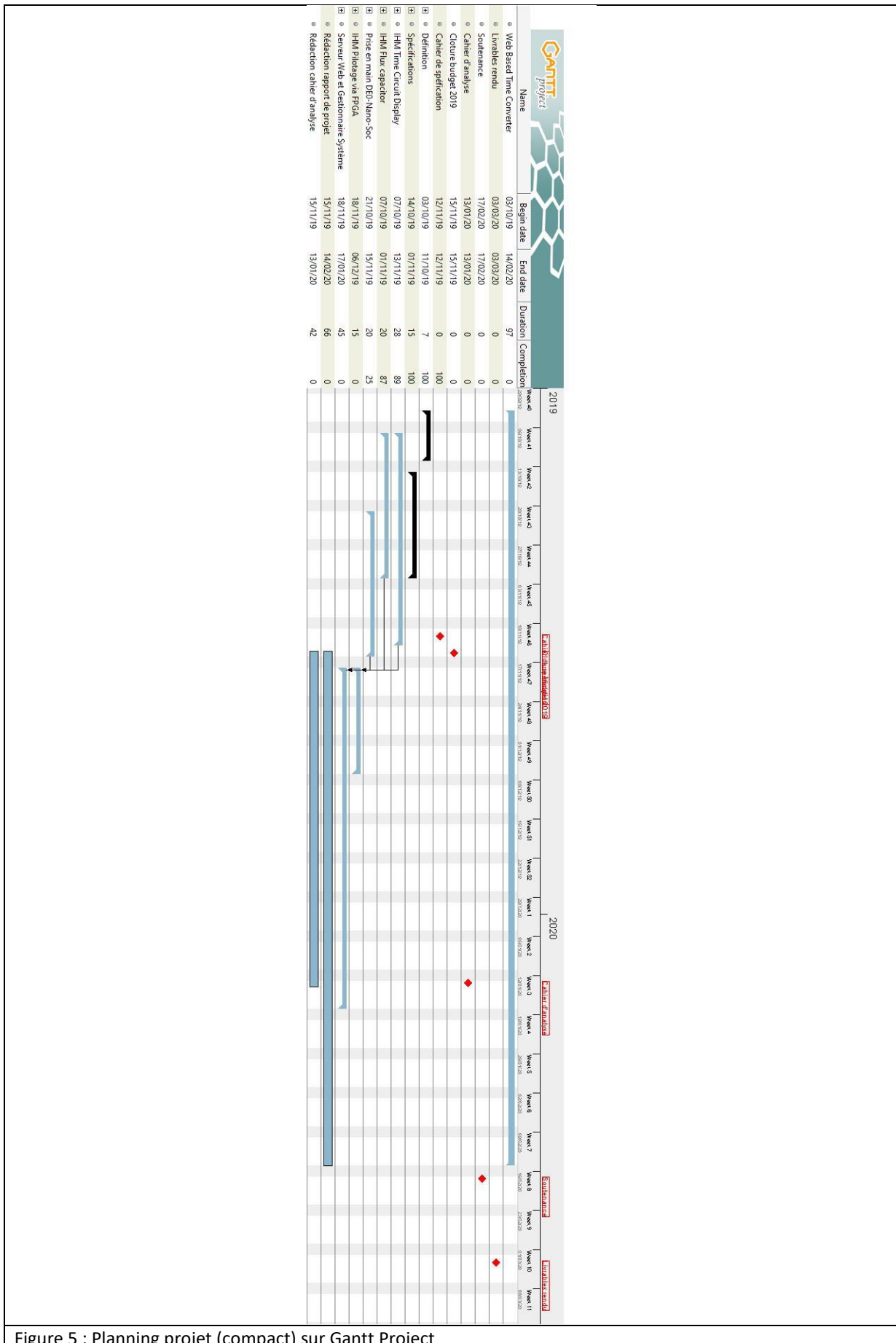


Figure 5 : Planning projet (compact) sur Gantt Project

## GLOSSAIRE

Dans cette partie on doit trouver, classés par ordre alphabétique, les définitions des termes courants utilisés, des termes techniques, abréviation, sigles et symboles employés dans l'ensemble du document.

**Altera EPCS** : Dispositif de chargement du « bit stream » d'un FPGA via un protocole série. C'est un composant matériel utilisé à la mise sous tension d'un FPGA.

**FPGA** : Composant logique programmable permettant de réaliser des circuits électroniques combinatoires ou séquentiel. Sa particularité est qu'il perd sa configuration à chaque coupure d'alimentation.

**HAL** : Hardware Abstraction Layer ou en français couche d'abstraction matérielle est une couche logicielle permettant l'accès au matériel par un jeu de routines de haut niveau.

**HPS** : Hard Processor System. C'est un processeur applicatif implanté dans un SOC à côté d'un FPGA. Contrairement au FPGA, l'architecture matérielle du HPS est figée dans le silicium, donc il n'y a pas moyen de le modifier matériellement. Autre points, dans la carte DE0-Nano-Soc, c'est un processeur ARM Cortex A9 double cœur qui sert de HPS.

**LED** : Light Emitting Diode, ou diode électroluminescente est un composant électronique pouvant émettre de la lumière sous l'effet du courant électrique qui la traverse. C'est un composant polarisé dans lequel le courant ne peut circuler que dans un sens à la façon d'une diode.

**PCB** : Printed Circuit Board. C'est un circuit imprimé associé en électronique à des composants pour créer une carte électronique. Les composants sont soudés sur le circuit imprimé.

**SOC** : System on Chip. C'est un circuit hautement intégré sur lequel peut fonctionner un système entier avec l'ensemble des éléments nécessaires à son fonctionnement (périphériques, calculateurs, circuits logiques, décodeurs...).

**TCD** : Time Circuit Display, c'est un des éléments phare du tableau de bord de la De Lorean de retours vers le futur. Le TCD affiche notamment l'heure et la date.

**VHDL** : Langage de description matériel permettant entre autres de synthétiser des fonctions électroniques dans un circuit logique programmable.

**WBTC** : Web based time converter. C'est le nom du projet.

## BIBLIOGRAPHIE

Aucune source spécifiée dans le document actif.

**Cahier de spécification 2018-2019** : A été utilisé pour la rédaction de ce document le cahier de spécification rédigé par Florian Rémy, premier étudiant sur ce projet en 2018-2019

**Time Circuit Display** : projet DIY de réplique du Time Circuit Display (images).

<https://www.partsnotincluded.com/propmaking/time-circuit-display-schematic/>

**Flux capacitor (convecteur temporel)** : Images et inspiration sur le convecteur temporel.

[https://backtothefuture.fandom.com/wiki/Flux\\_capacitor](https://backtothefuture.fandom.com/wiki/Flux_capacitor)

## INDEX

Cette partie indique les pages où sont traités et mentionnés les sujets et les termes les plus importants du document.

**Aucune entrée d'index n'a été trouvée.**