Cahier des CHARGES : projet DEV groupe 34

Deuxième semestre, année 2016-2017

Gardin Alan, Hachicha Anis, Kheder Abdessalem

Encadrant : Guillaume Le Gall

Table des matières

[Table des matières 1](#_Toc476558663)

[1 Contexte et définition du problème 1](#_Toc476558666)

[1.1 Terminologie, abréviations 1](#_Toc476558667)

[1.2 Licence 2](#_Toc476558670)

[2 Spécifications fonctionnelles et techniques 2](#_Toc476558671)

[2.1 Livrables 2](#_Toc476558672)

[2.2 Technologies utilisées 2](#_Toc476558673)

[2.3 Hiérarchisation des ressources serveurs 2](#_Toc476558674)

[2.3.1 Serveur EC 2](#_Toc476558675)

[2.3.2 Serveur BMS 2](#_Toc476558676)

[2.4 Exigences fonctionnelles 3](#_Toc476558677)

[2.4.1 Fonctionnalités côté serveur 4](#_Toc476558678)

[2.5 Exigences non-fonctionnelles 5](#_Toc476558679)

[2.5.1 Plateforme de tests 5](#_Toc476558680)

[2.5.2 Rendement 5](#_Toc476558681)

[2.5.3 Maintenabilité 5](#_Toc476558682)

[2.5.4 Portabilité 5](#_Toc476558683)

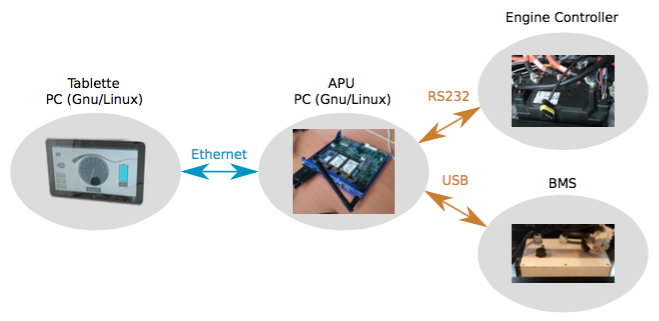
[3 Synthèses des exigences 5](#_Toc476558684)

[3.1 Exigences fonctionnelles 5](#_Toc476558685)

[3.2 Exigences non-fonctionnelles 5](#_Toc476558686)

[4 Références 5](#_Toc476558687)

# Contexte et définition du problème

Sur le campus de Rennes, une équipe de chercheurs travaille sur un véhicule électrique Open-source. Le moteur et les batteries du véhicule sont respectivement reliés à un contrôleur de moteur « EC » et un système de contrôle de batteries « BMS » qui fournissent plusieurs données à un système embarqué « APU ». D’autre part, une application « tableau de bord » pour tablette a été développée. Reste à établir la connexion entre l’APU et la tablette pour que le conducteur puisse accéder à toutes les données concernant son véhicule. C’est l’objet de ce projet. Ce document décrit les fonctionnalités et les contraintes qui devront être respectées par le prototype

(Guillaume Le Gall)

## Terminologie, abréviations

Californium : Libraire Java permettant d’utiliser le protocole CoAP.

JSon : Format de représentation de données.

JSon-Simple : Bibliothèque Java permettant une utilisation simple du format JSon.

OSV : Open Source Vehicule (osvehicle.com)

BMS : Battery Management System.

EC : Engine Controller.

APU : Système embarqué présent sur le véhicule. (pcengines.ch/apu.htm)

VIT : Travail absolument requis.

MIN : Travail souhaité mais non nécessaire.

## Licence

Le travail sera effectué sous licence GPLv3 dans la continuation du travail déjà effectué par le client. Les modalités sont explicitées ici : www.gnu.org/licenses/quick-guide-gplv3.html

# Spécifications fonctionnelles et techniques

## Livrables

Les livrables sont attendus pour la fin du deuxième semestre de l’année 2016-2017. Il comprendra plusieurs codes sources, des programmes de tests et d’exemple, ainsi que la documentation du code au format Javadoc. En particulier, trois applications sont attendues :

* Une application serveur qui fournit les ressources du BMS.
* Une application serveur qui fournit les ressources du EC.
* Une application client qui reçoit les données des serveurs et les affiche sur la tablette.

Un prototype de fichier Vagrant décrivant un environnement virtuel de test est souhaité.

## Technologies utilisées

Les programmes livrés seront écrits à l’aide du langage Java. Le format retenu pour les ressources est le format JSon. Le programme client communiquera avec les programmes serveur par l’intermédiaire du protocole CoAP. Les programmes utilisés utiliseront les bibliothèques Californium et JSon-Simple.

## Hiérarchisation des ressources serveurs

### Serveur EC

* /speed
* /kms

### Serveur BMS

* /soc
* /charging
* /cell\_voltage
* /cell\_voltage/min
* /cell\_voltage/max
* /cell\_voltage/average
* /voltage
* /current
* /power

## Exigences fonctionnelles

L’objectif est d’assurer la communication de bout en bout entre la tablette et l’APU.

* + 1. Fonctionnalités côté client

|  |  |
| --- | --- |
| **Nom** | **Initialisation Client** |
| Description | Permet à une application client de démarrer. |
| Evénement déclencheur | Allumage de la tablette |
| Entrées | Aucune |
| Sortie | Erreur si échec |
| Contraintes | La tablette était préalablement connectée au réseau. |
| Importance | VIT |

|  |  |
| --- | --- |
| **Nom** | **Demande d’un ressource serveur** |
| Description | Permet à la tablette de demander une ressource à un serveur. |
| Evénement déclencheur | Demande explicite |
| Entrées | AdresseServeur ET Port ET TypeRequête ET NomRessource |
| Sortie | Erreur si le serveur est introuvable.  Erreur si la ressource demandée est introuvable  La ressource au format JSon sinon |
| Contraintes | La tablette était préalablement connectée au réseau.  L’adresse est au format IPv6.  La ressource fournie est au format JSon |
| Importance | VIT |

|  |  |
| --- | --- |
| **Nom** | **Découverte des ressources serveur** |
| Description | Permet à la tablette de découvrir les ressources disponibles sur un serveur (requête GET pour l’index) |
| Evénement déclencheur | Demande explicite |
| Entrées | AdresseServeur ET Port |
| Sortie | Erreur si la tablette n’était pas connectée au réseau.  Erreur sur le serveur est introuvable.  Liste des ressources au format JSon sinon. |
| Contraintes | La tablette était préalablement connectée au réseau. |
| Importance | MIN |

|  |  |
| --- | --- |
| **Nom** | **Découverte des autres équipements** |
| Description | Permet à la tablette de découvrir les autres équipements présents sur le réseau. |
| Evénement déclencheur | Demande explicite |
| Entrées | Aucune. |
| Sortie | Erreur si la tablette n’était pas connectée au réseau.  Liste des adresses IPv6 des autres équipements, numéro de ports, et description sinon. |
| Contraintes | La tablette était préalablement connectée au réseau. |
| Importance | MIN |

### Fonctionnalités côté serveur

|  |  |
| --- | --- |
| **Nom** | **Démarrage d’un serveur** |
| Description | Permet à une application serveur de démarrer le serveur |
| Evénement déclencheur | Démarrage tablette |
| Entrées | Port |
| Sortie | Erreur si échec |
| Contraintes | Serveur CoAP  Ressources au format JSon |
| Importance | VIT |

|  |  |
| --- | --- |
| **Nom** | **Répondre à une requête** |
| Description | Permet à une application serveur de répondre à une requête client. |
| Evénement déclencheur | Requête client. |
| Entrées | TypeRequête ET nomRessource (ET paramètres optionnels) |
| Sortie | Erreur si ressource introuvable.  Message de succès si requête POST et succès.  Ressource au format JSon si requête GET et succès. |
| Contraintes | La ressource est fournie au format JSon |
| Importance | VIT |

|  |  |
| --- | --- |
| **Nom** | **Arrêt d’un serveur** |
| Description | Permet à une application serveur d’arrêter serveur |
| Evénement déclencheur | Demande explicite |
| Entrées | Aucune |
| Sortie | Erreur si échec  Erreur si serveur non démarré. |
| Contraintes | Le serveur doit avoir été démarré. |
| Importance | MIN |

## Exigences non-fonctionnelles

### Plateforme de tests

Un environnement de test peut être proposé. Il s’agirait, à l’aide deux images virtuelles (tablette et APU), de créer un environnement de simulation déployable rapidement. Une telle solution devra utiliser Vagrant, pour la gestion des images virtuelles et le déploiement.

### Rendement

Les serveurs CoAP doivent pouvoir répondre aux requêtes dans un délai raisonnable. Cet effort d’optimisation est d’autant plus important dans le cas des systèmes embarqués.

### Maintenabilité

Le code source fournit sera commenté au format Javadoc, et accompagné de programmes d’exemples. L’environnement de test fournira ainsi une solution portable et universelle pour les futurs tests de développement.

### Portabilité

Les applications fournies devront être compatible avec n’importe quel environnement Java similaire à celui de la tablette, sous réserve que celui-ci soit doté d’une architecture réseau semblable à celle de l’OSV.

# Synthèses des exigences

## Exigences fonctionnelles

|  |  |
| --- | --- |
| **Exigence** | **Importance** |
| Initialisation Client | VIT |
| Demande d’un ressource serveur | VIT |
| Découverte des ressources serveur | MIN |
| Découverte des autres équipements | MIN |
| Démarrage d’un serveur | VIT |
| Répondre à une requête | VIT |
| Arrêt d’un serveur | MIN |

## Exigences non-fonctionnelles

|  |  |
| --- | --- |
| **Exigence** | **Importance** |
| Rendement | VIT |
| Maintenabilité | VIT |
| Portabilité | VIT |
| Plateforme de test | MIN |

# Références

<https://www.osvehicle.com/>

<http://pcengines.ch/apu.htm>

[www.gnu.org/licenses/quick-guide-gplv3.html](http://www.gnu.org/licenses/quick-guide-gplv3.html)