



Compte-rendu de réunion

Compte-rendu n°2
Date : 10/02/23

Rencontre du client / ProSE

Appel à l'ordre :

Date de la réunion : 09/02 **Heure début et fin :** 10h30 -> 12h30 **Lieu :** ESEO-Antarctique

Réunion organisée par : Camille CONSTANT Type de réunion : Clarification du cahier des charges Rédacteur : Gabriel MARQUETTE, Camille LENNE Point météo : Ensoleillé	Participants : Toute l'équipe ProSE, Alain RIBAUT, Camille CONSTANT
---	--

Objectif de la réunion :

Cette réunion était destinée à déterminer les exigences du projet. Le but principal était de clarifier le cahier des charges et de poser toutes les questions nécessaires pour mieux comprendre le sujet. L'objectif était d'obtenir une image complète et détaillée du projet.

Sujets abordés et actions menées :

- Utilisation d'un Template pour les livrables avec un suivi régulier (1 fois par semaine)
- Technique
 - o Capteur / Actionneurs : clignotants, warnings, affichage vitesse, etc.
 - o Carte électronique : Raspberry fourni par Matthias
 - o Clarification de l'architecture logicielle et matérielle
 - o Passerelle physique, pas d'implémentation de driver
- IHM – Android
 - o Besoin : Envoi/Réception de trames
 - o Élaboration d'une première maquette avec le client
 - o Message d'erreur compréhensible par n'importe qui

Commentaires :

La première rencontre avec Alain RIBAUT était limpide et efficace.

Description plus détaillée de la réunion

Compte Rendu (CR) de la réunion du 9 février - *Rencontre avec le client*

Table des matières

1.	CBOE	2
2.	LIVRABLES ET SUIVI.....	2
3.	TECHNIQUE	3
4.	IHM - ANDROID	4
5.	ÉCO-CONCEPTION.....	5
6.	DEVELOPPEMENT SECURISE	6
7.	TESTS	6
8.	CULTURE GENERALE.....	6
9.	AUTRE.....	6

1. CBOE

Besoin : Éviter d'emporter le banc de test lorsque l'entreprise se déplace dans une école. Le simulateur ainsi qu'un Raspberry doit être suffisant. Permet également de visualiser quelque chose de concret pour les personnes novices dans le métier (nouveau salarié, élèves...) et ainsi permettre une montée en compétences sur l'architecture véhicule, le réseau CAN ainsi que d'un démonstrateur CAN.

KEREVAL souhaite également pouvoir faire une injection de faute via les trames pour vérifier que cela fonctionne comme cela devrait.

Pas une réponse à un problème mais plus un besoin d'une amélioration.

2. Livrables et suivi

a. Livrables

On utilise les Template que l'on souhaite. Donc Template à harmoniser avec les autres groupes. Le fond importe plus que la forme.

b. Suivi

Un point régulier une fois par semaine dans un premier temps, fréquence pour la suite à voir avec KEREVAL. Une réunion via Teams pour des points sont possibles dans le canal créé par Théo BENARD.

Pour le contenu des points, un point météo ainsi qu'un mail qui résume les activités majeures avec des questions si nécessaires avec quelques détails conviennent dans un premier temps. A terme, KEREVAL souhaite que chaque exigence soit contrôlée et vérifiée par un test.

3. Technique

a. Banc de test

Le banc de test est similaire à la photo disponible dans le cahier des charges, page 5. Il permet des expériences de test grâce à la présence de calculateurs, il permet aussi l'injection de fautes sur le bus can et la vérification pour prendre le contrôle sur le véhicule.

Il s'agit de matériel récupéré à la casse ou chez des fournisseurs. Cela permet de simuler le fonctionnement d'une voiture. Architecture PSA, Renault ou Volkswagen. Dans notre cas, le banc n'est pas de type PSA, il est spécifique.

Pas de communication ou de documentation sur le banc mais possibilité de revenir voir le client si besoin ou difficultés. De même si blocage.

c. Capteurs & Actionneurs

Il possède :

- Clignotants (Comodo)
- Affichage de la vitesse
- Warnings
- Afficheurs pour les commandes

d. Carte Électronique

Raspberry fourni par Matthias BRUN ou Camille CONSTANT. Si problème il y a, KEREVAL peut nous en fournir une.

e. DBC

Présence de la DBC. Accès à la DBC via BusMaster. Redemander au client si besoin.

f. Information inconnue

Connexion simulateur sur le bus CAN pour lever le risque. Le simulateur se branche actuellement sur une interface simulé CAN et le client souhaite le passé en physique.

g. Application évolutive

L'information d'un composant ne doit pas être codé en dur (Ajout potentiel d'un volant dans le futur par exemple). L'application Android doit permettre d'écrire et envoyer des trames de n'importe quel type. (Possibilité de les modifier, copier etc.).

h. Passerelle CAN & Cycle d'allumage

Passerelle physique, pas d'implémentation de driver. Shield CAN natif. Cycle d'allumage du banc de test : pas besoin de cycle d'allumage (à revoir avec les experts).

CAN High Speed et CAN Low Speed correspondent à la vitesse de réaction.

KEREVAL possède potentiellement un fichier extra CAN, à voir avec Yannick pour le document complémentaire.

i. Fonctionnement

Connexion du portable par wifi sur la Raspberry et connexion par réseau can entre la Raspberry et le simulateur/banc de test. On peut passer par le PEAK.

Le portable doit donc être connecté au simulateur et au banc mais pas directement (présence du système embarqué entre les deux).

j. Schéma

APP Android --> Smartphone --> WIFI (Bluetooth optionnel) --> Raspberry (Add-on branché dessus) --> Réseau CAN --> Simulateur/Banc de test

Nécessité d'afficher la même donnée sur le simulateur et le banc de test. Le banc a l'aspect habitacle/aspects moteur sur le simulateur.

k. Plan d'action recommandé

Essayer de connecter la Raspberry et le PC pour valider la connexion.

Avec 2 outils de relevé CAN : tester la connexion entre la Raspberry et le PC. Ajout du simulateur pour assurer la connexion. Raspberry --> Simulateur. Remplacer Linux par Simulateur.

Mapping du simulateur pour regarder si le banc de test bouge.

- 1 - Faire bouger des choses sur le simulateur
- 2 - Faire bouger des choses sur le banc de test

Le réel intérêt est de faire bouger les choses sur le simulateur et sur le banc de test depuis le portable. La même trame n'est pas nécessaire pour obtenir le résultat.

l. Système d'exploitation

Système d'exploitation récent : Moins d'un an à compter du début du projet.

b. IHM - Android

a. Besoin

Envoi/Réception de trames.

Pencil (Soft) qui permet de décrire notre application et on reviendra vers le client pour confirmer ou modifier.

L'application est en format portrait. La partie responsive est optionnelle.

b. Partie haute de l'écran

- Envoi & Réception des trames.
- Connaître l'objet que l'on pilote. Afficher l'objet selon une arborescence.

- Nombres de trames affiché fixe mais on peut obtenir plus d'informations avec un clic sur le "+". Cela fonctionne comme un "voir plus" sur les RS. Cela ne prend pas plus de place, juste un bar pour scroller et ainsi éviter de polluer la vue de l'écran si y'a beaucoup de trames.
- Un seul objet ouvert à la fois. Si un objet est ouvert, lors de l'ouverture d'un nouveau, cela va fermer le précédent.
- Icône en haut pour créer, envoyer et enregistrer une nouvelle trame.
- Lors de l'installation, il n'y a pas d'objet déjà enregistré.
- Les objets ainsi que les trames sont persistants.
- Envoi de trames de manière cyclique ou ponctuelle via un bouton à choix unique.

c. Partie basse de l'écran

Terminal en temps réel qui va afficher toutes les trames reçues.

Infos des trames à afficher :

- Trames
- KEREVAL doit nous faire un retour sur infos à afficher par les trames

d. Idée abandonnée

Filtre avec des traces correspondante sur l'écran du bas.

e. Informations supplémentaires

- Envoie et réception dans la même case
- Envoie en cyclique ou une seule fois
- Besoin ou pas de tram étendu : retour nécessaire du client

f. Erreurs

Message d'erreur compréhensible par n'importe qui.

Afficher un pop-up, corriger l'erreur via un doc à coté ou directement sur le pop-up.
Pas de préférence du client.

c. Éco-Conception

L'objectif de l'empreinte carbone est plutôt une sensibilisation à l'éco-conception.

L'idéal serait de marquer pour chaque élément optionnel (nombre d'heures / coût / état actuel et axes d'améliorations).

La puissance de la consommation maximale doit nous être donné par KEREVAL.

d. Développement sécurisé

L'application ou la passerelle doivent être authentifiées, le choix est de notre côté. (Si je possède l'application, j'ai le droit de me connecter à une passerelle).

Il n'est pas possible d'avoir plusieurs connexions de l'application.

e. Tests

Arrêt/redémarrage : sauvegarde prévue intrinsèquement au niveau de l'architecture ; tout est sauvegardé par la Raspberry normalement.

Présence de tests pour le 3.1 pour montrer que l'exigence est respectée.

f. Culture Générale

Prise OBD programmable (possibilité de plus d'informations lors de la visite à Rennes).

g. Autre

Yannick et Kylian : Expert technique de chez KEREVAL.