



# MASTER 2 – ISIDIS

## Projet de synthèse

---

V.A.L.

---

**Vision Doc**

---

## Suivi de versions :

Révision	Date	Rédacteur(s)	Nature de la modification
0	15/11/2012	BOUTABA RAFIK	Version Pré-réunion méthodologique 1
1	20/11/2012	BOUTABA RAFIK	Version Post-réunion méthodologique 1
2	28/11/2012	BOUTABA RAFIK	Version Pré-réunion méthodologique 2
3	01/12/2012	BOUTABA RAFIK	Version Post-réunion méthodologique 2
4	02/12/2012	FATAL-TEAM	Version Comité de Pilotage 1

## Table des matières

<b>1. Objet du document</b>	<b>4</b>
<b>2. Objet du projet</b>	<b>4</b>
<b>3. Objectifs et enjeux</b>	<b>4</b>
<b>4. Positionnement</b>	<b>5</b>
4.1 Positionnement du probleme	5
4.2 Positionnement commercial	5
<b>5. DETERMINER LES UTILISATEURS ET DECIDEURS</b>	<b>5</b>
5.1 UTILISATEURS:	5
5.2 DECIDEURS:	6
<b>6. LIENS ENTRE APPLICATIONS</b>	<b>6</b>
<b>7. FONCTIONNALITES DU PROJET</b>	<b>7</b>
<b>8. EXIGENCES ET CONTRAINTES SUPPLEMENTAIRES</b>	<b>7</b>
8.1 EXIGENCES <<FURPS>>	7
8.1.1 Functionality:	7
8.1.2 Usability:	7
8.1.3 Reliability:	7
8.1.4 Performance:	8
8.1.5 Supportability:	8
8.2 Autres contraintes	8
8.2.1 Documentation	8
8.2.2 Organisation	8
8.2.3 Temps	8

Documents de référence :

- Document sourceforge sur le vision document : [Lien](#)
- Sujet du projet de synthèse
- Cours ESIAG

## **1. Objet du document**

Ce document a pour objectif d'identifier les décideurs et les utilisateurs affectés par le problème impliquant ce projet et la définition du périmètre du projet à réaliser afin d'en cadrer les besoins métier. Il a également pour vocation de présenter une vue globale (haut niveau) de la solution à réaliser. Enfin, ce document contient un aperçu des exigences et contraintes recensées ce qui constitue une base contractuelle.

## **2. Objet du projet (fonction du projet)**

Le projet de synthèse sera mené par une équipe de travail de dix étudiants Miagistes et aura comme but principal la réalisation d'une solution performante, respectant les caractéristiques des systèmes distribués, et fiable.

Le projet a pour objectif la réalisation d'une application destinée à une entreprise de transports en commun exploitant un réseau ferré et des équipements automatiques assurant le transport de passagers sur un secteur géographique à définir, ainsi que l'étendue de son infrastructure (quais, couloirs, tunnels, rames, appareils de détection .. etc)

La solution logicielle doit être sûre et ce en l'utilisant via un environnement de simulation dynamique, tout en étant ouverte et évolutive pour être implantée dans toute entreprise du secteur.

## **3. Objectif et enjeux**

Le but du projet est de fournir un SI permettant l'automatisation du réseau client, et permettre son pilotage en temps réel.

Le pilotage automatique implique la présence d'un centre de contrôle faisant la liaison entre les interfaces de communication et le réseau terrain afin de superviser toute l'activité du réseau, et de ce fait pouvoir interagir avec ce dernier en temps réel.

Les moyens de communication doivent donc être sûrs, et donc très hautement disponibles (contrôle de charge)

## **4. Positionnement**

### **4.1 Positionnement du problème**

<b>Problème</b>	Le SI actuel ne permet pas la gestion automatique en temps réel du réseau
<b>Affecte</b>	L'exploitation du réseau, les conditions de supervision
<b>Conséquence</b>	sur le trafic, et les performances

<b>Solution</b>	superviser la totalité du reseau, réactivité en cas d'incident
-----------------	----------------------------------------------------------------

## 4.2 Positionnement commercial

<b>Pour</b>	sociétés de transport rencontrant des problèmes d'exploitation de leur reseau de transport
<b>Qui (besoin)</b>	supervise l'ensemble de ses activités, en capturant/priorisant/transmettant les informations terrain en cas d'incident afin de réguler le trafic.
<b>RTDG (fatal-team)</b>	interconnexion entre plusieurs composants dédiés à des tâches précises, adaptés aux contraintes clientes pour résoudre ses problèmes
<b>Assurant</b>	la haute disponibilité de l'interconnexion entre le centre de controle et les équipements terrain, la remontée d'information en temps réel selon un ordre de priorité défini.
<b>Valeur ajoutée</b>	par rapport a l'existant, notre solution permet une supervision globale du système, d'avoir une remontée des alertes terrain ET l'envoi d'ordre en cas d'incident grave
<b>Le produit</b>	repond au problèmes récurrents liés a l'exploitation des reseaux de transport ferrés, tout en s'adaptant au reseau d'un client particulier

## 5. DETERMINER LES UTILISATEURS ET DECIDEURS

### 5.1 Utilisateurs

Nous avons identifié deux types d'acteurs en fonction du rôle qu'ils jouent :

- Administrateur RTDG (Superviseur) : il se charge de démarer et d'arreter le composant RTDG
- Responsable RTDRS : utilise le RTDGs

### 5.2 Decideurs

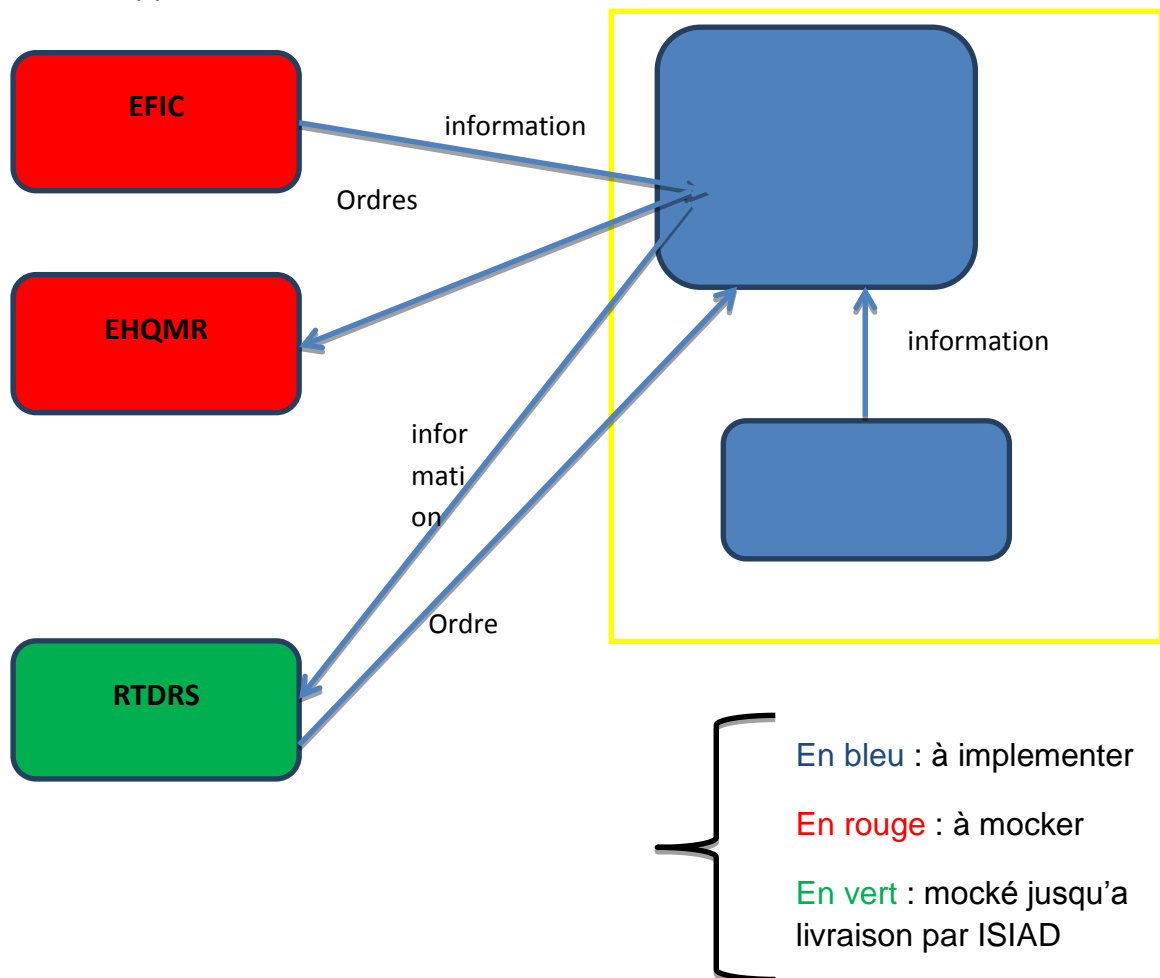
- Equipe pédagogique : (Alexandre Brenner et Gilles Giraud)
- Mode d'implication: Comité de pilotage.

- Critère de succès : Robustesse, performance, transparence, simplicité, clarté, et respect des normes.

- Product Owner : son rôle est de représenter auprès de l'équipe de développement le client/utilisateur pour que le produit final soit conforme aux attentes de ce dernier auprès de l'équipe de développement, revue de sprint et backlog, et accepter ou refuser des un travail réalisé de par sa connaissance du domaine métier, et ce au moyen de réunions ponctuelles en fonction de sa disponibilité
- Responsable RTDRS : doit nous indiquer la manière dont on communique avec le système, et se mettre d'accord sur les informations qui doivent être échangés entre le RTDRS et notre RTDG.

## 6. Liens entre applications

les différents composants utilisés dans notre solutions communiquant sont interconnectés (via l'envoi de messages), le schéma ci dessous presente une vue globale de l'application :



## 7. FONCTIONNALITES DU PROJET

Nom	Description	Priorité
<b>Prioriser messages recus</b>	Associer aux messages recus par les composants terrain un ordre de priorité croissant (de la plus critique à la moins critique	Haute
<b>envoyer les informations</b>	envoyer les informations capturées par les équipements terrain au RTDG	Haute
<b>recevoir les informations</b>	recevoir les informations capturées par les équipements terrain	Haute
<b>Envoyer les ordres</b>	transmettre les ordres en provenance du centre de controle au terrain	Haute
<b>Recevoir les ordres</b>	recevoir les ordres en provenance du centre de controle	Haute
<b>Sauvegarder les informations</b>	sauvegarder les information et les ordres transmis	Moyenne
<b>Gerer le RTDG</b>	superviser l'ensemble des activités du RTDG, les informations et les ordres	Moyenne

## 8. EXIGEANCES ET CONTRAINTES SUPPLEMENTAIRES

### 8.1 Exigences FURPS

#### 8.1.1 Functionality

- la synchronisation des données doit être faite de manière à ce que la perte de données soit la plus minimale possible.
- La modification à chaud des valeurs de configuration.

#### 8.1.2 Usability

- Le comportement de l'application change dynamiquement selon le scenario qu'on lui fournit en entrée dans le Framework de simulation developpé.
- 

#### 8.1.3 Reliability

- Notre application se doit d'être tolerante aux pannes c'est à dire qu'elle doit continuer a fonctionner normalement ( ou du moins légèrement dégradée )

lorsqu'un de ses composants ne fonctionne pas correctement, ceci est assuré notamment en faisant de la réplication/duplication des composants fortement sollicités

- Développement d'un environnement de simulation dynamique Garantir aux clients la validité de la solution logicielle et ce notamment en utilisant.

#### **8.1.4 Performance**

- Le critère performance est un atout majeur de notre application, la remontée des informations ainsi que la transmission des messages d'ordre doit être fait en temps quasi réel , c'est à dire dès leur production dans les composants embarqués
- Elle doit aussi assurer une disponibilité élevée ce qui implique la mise en oeuvre d'algorithmes de répartition de charge pour distribuer le travail entre les replications du composant RTDG qu'on créé
- Le système doit pouvoir supporter une certaine volumétrie (qui peut être importante) de données, informations capturées sur le terrain, messages échangés entre les différents composants, messages d'ordres .. etc

#### **8.1.5 Supportability**

- Notre application doit être adaptable à tout réseau client de même périmètre applicatif (nombre de stations, de lignes, ...etc )
- Le déploiement doit être automatisé autant que possible, ou du moins avec le moins de paramétrage manuels possible

### **8.2 Autre contraintes**

#### **8.2.1 Documentation**

On va définir une documentation afin d'aider l'utilisateur à bien utiliser et gérer le produit.

#### **8.2.2 Organisation**

Le projet s'inscrit dans le cadre d'un projet universitaire, l'intégration entre les membres du groupe est préliminaire par rapport au sujet, il faut prendre en considération la disponibilité et le lieu du travail de l'équipe pour la distribution des tâches qui seront réalisées dans ce cas à distance

L'impact contraintes liés à l'organisation de travail est en quelque sorte diminué grâce à la mise en place d'outils et logiciels facilitant la communication et le travail à distance (gestionnaire de version, gestionnaire de conduite de projet, etc.)

En outre une partie de l'application doit être réalisée par une équipe d'une autre formation de l'ESIAG, équipe ISIAD doit nous fournir le composant RTDRS qui nous fournit notamment le horaires des trains ..etc

#### **8.2.3 Temps**

Le temps de la réalisation de chaque tâche est important, il faut alors respecter les délais estimés pour chaque tâche afin de les tester et d'optimiser le temps de la réalisation car c'est un critère fondamentale à respecter pour l'état d'avancement du projet.