



# Sommaire

- PRÉSENTATION GÉNÉRALE

- Expression du Besoin
- Présentation du projet
- Diagramme de déploiement
- Présentation des modules
- Répartition des tâches
- Diagramme de cas d'utilisation
- Planification
- Les ressources de développement

- PRÉSENTATION PERSONNELLE

- **Logiciel Chrono-Cross**

- Diagramme de classe
    - Diagramme de base de donnée

- Communication

- Configuration du port RS232

- Protocole THCOM08

- **Logiciel Gestion-Cross**

- Diagramme de classe
    - Diagramme de base de donnée

- Fonctionnalités de l'IHM

- Les tests et améliorations possibles

# Présentation générale

# Expression du besoin

- Equipe enseignants d'EPS
- Système de chronométrage et de classement de course de cross
- Simple d'utilisation
- Un délai de 200 heures

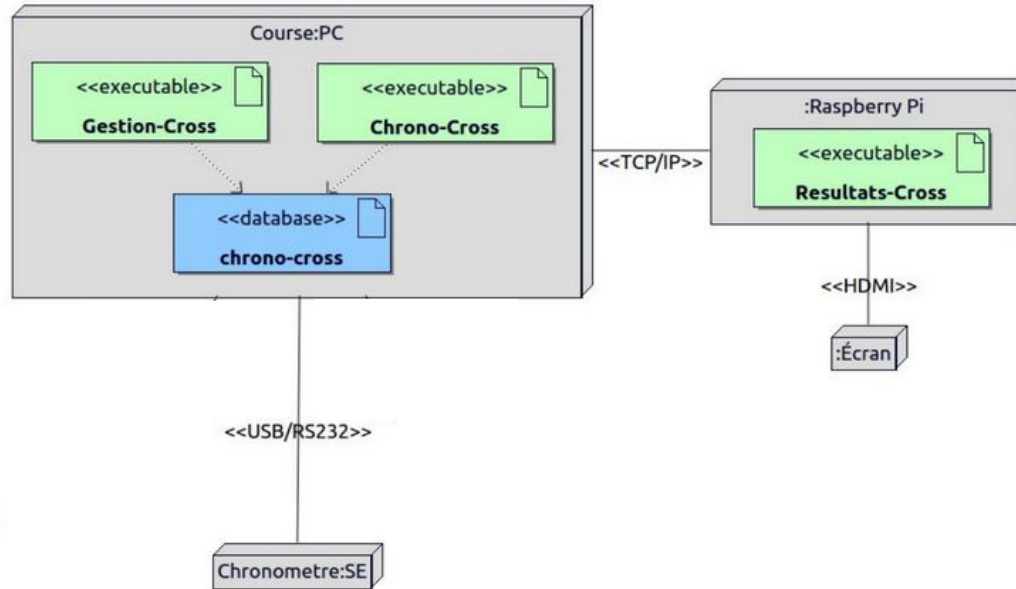
**Gérer des courses de cross chronométrées pour un établissement scolaire**

# Présentation du projet

- Système informatisé de gestion de course à pieds composé des modules :
  - Chrono - Cross
  - Gestion - Cross
  - Résultat - Cross
  - Un module de chronométrage (et de détection de coureurs)



# Diagramme de déploiement



# Présentation des modules

## Chrono - Cross

- Gérer une course
- Lancer et arrêter le chronomètre
- Classer les arrivés

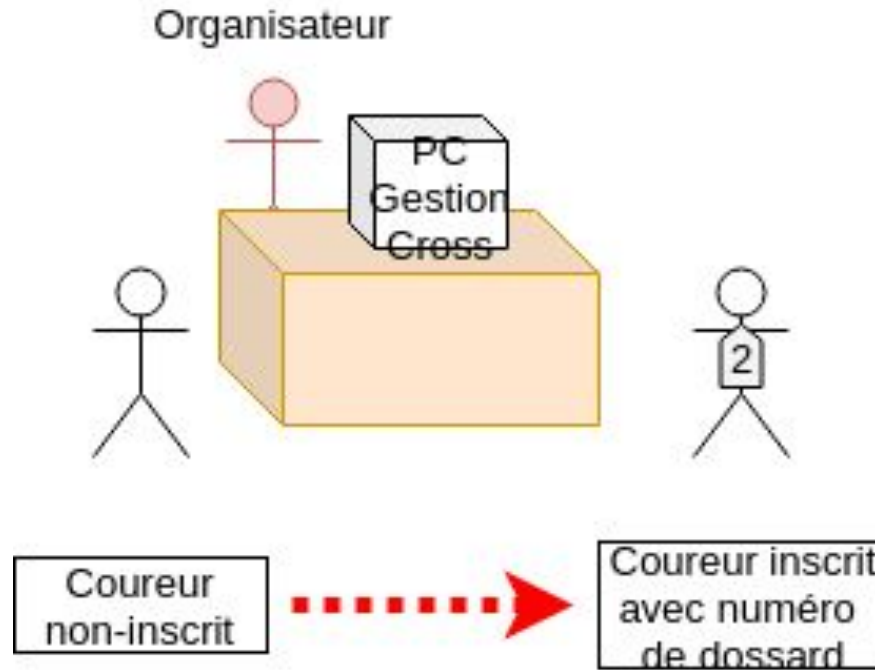
## Resultats - Cross

- Visualiser les classement et les temps
- Imprimer les résultats une fois la course fini

## Gestion - Cross

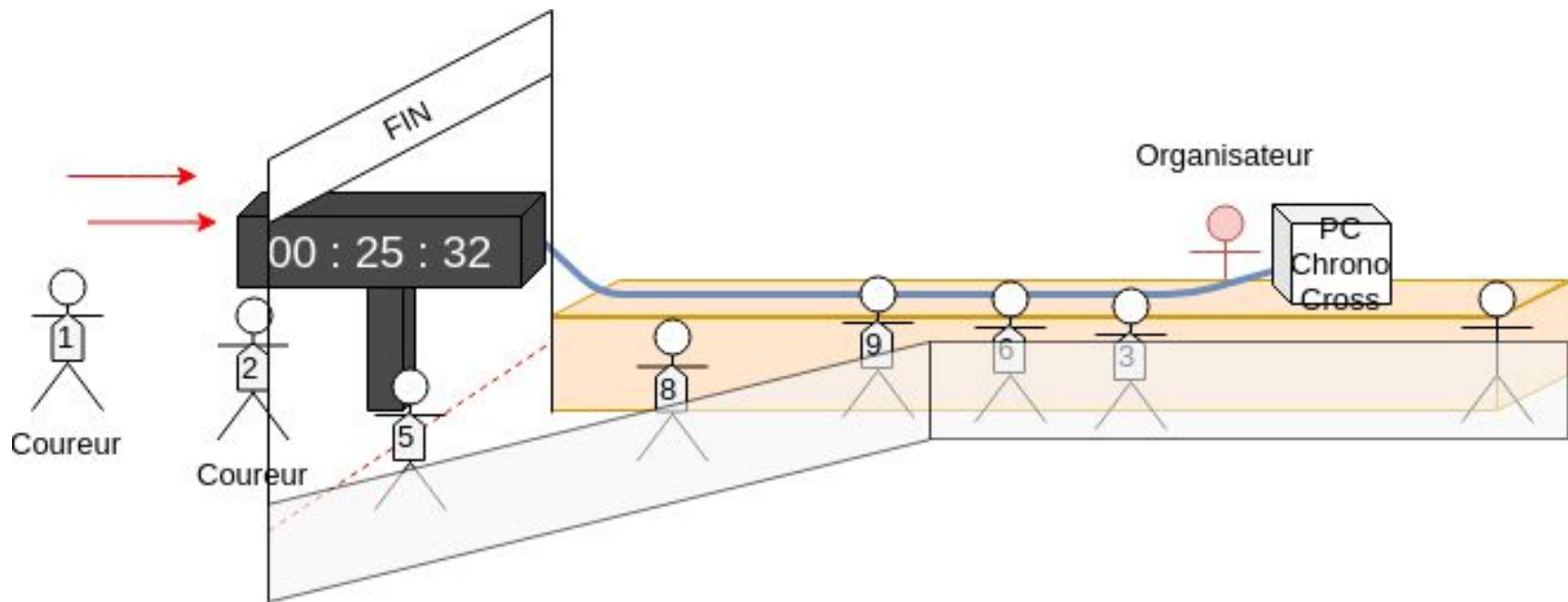
- Éditer une **course** / une **manifestation**
- Editer et inscrire un coureur à une course en lui attribuant un dossard

# Présentation du module Gestion-Cross





# Présentation du module Chrono-Cross



# Présentation du module de chronométrage

## Module de chronométrage

- Afficheur matriciel à leds
- Liaison série RS-232
- Protocole THCOM08
- 2 Canaux pour l'infrarouge

## Module de détection des coureurs

- Simulateur de cellule infrarouge d'arrivée



*TAG HEUER HL 975*

# Répartition des tâches - Étudiant 3 IR

## Installation

- La Base de données

## Configuration

- Communication
- Base de données

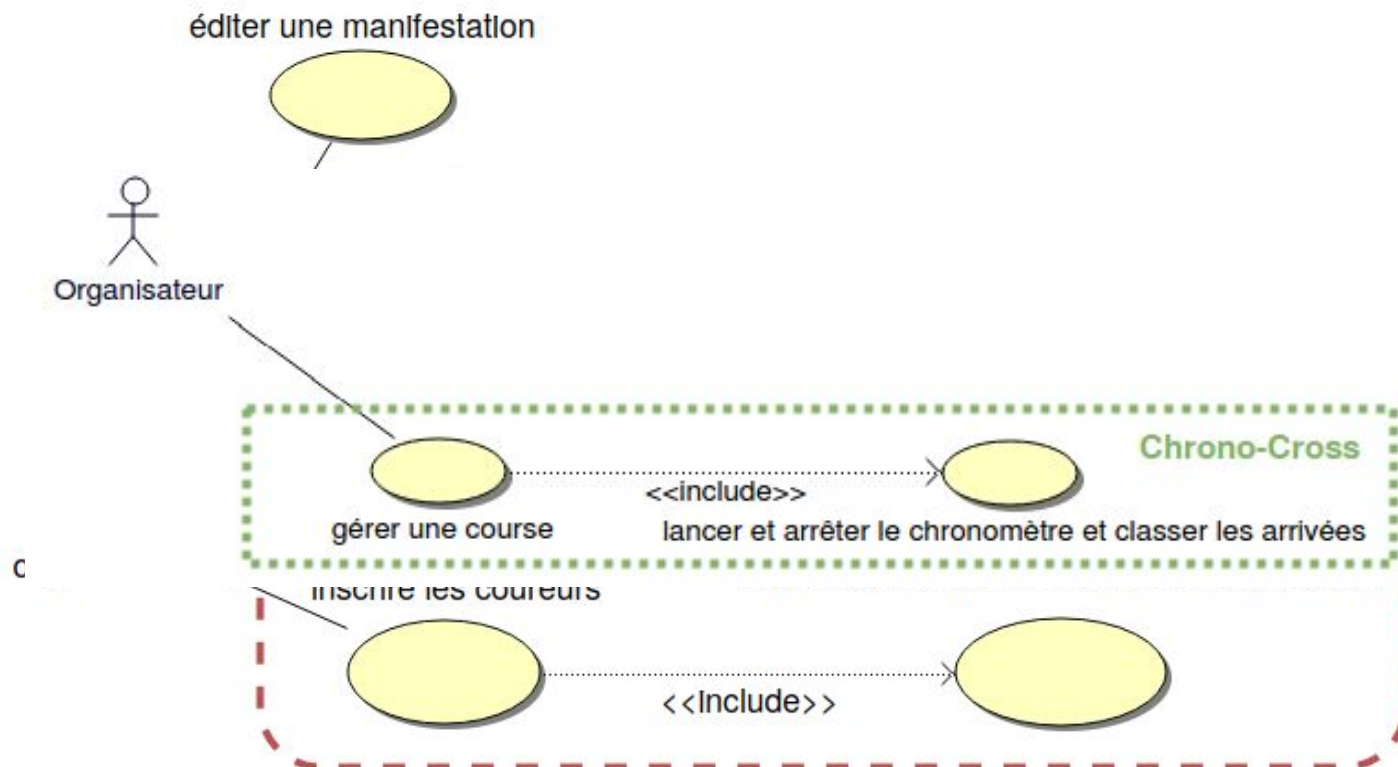
## Cas d'utilisation

- Éditer les coureurs
- Inscrire les coureurs
- Démarrer une course
- Lancer et arrêter un chronomètre
- Classer les arrivées

ANDRÉO Michaël

*Chrono - Cross*  
*Gestion - Cross*

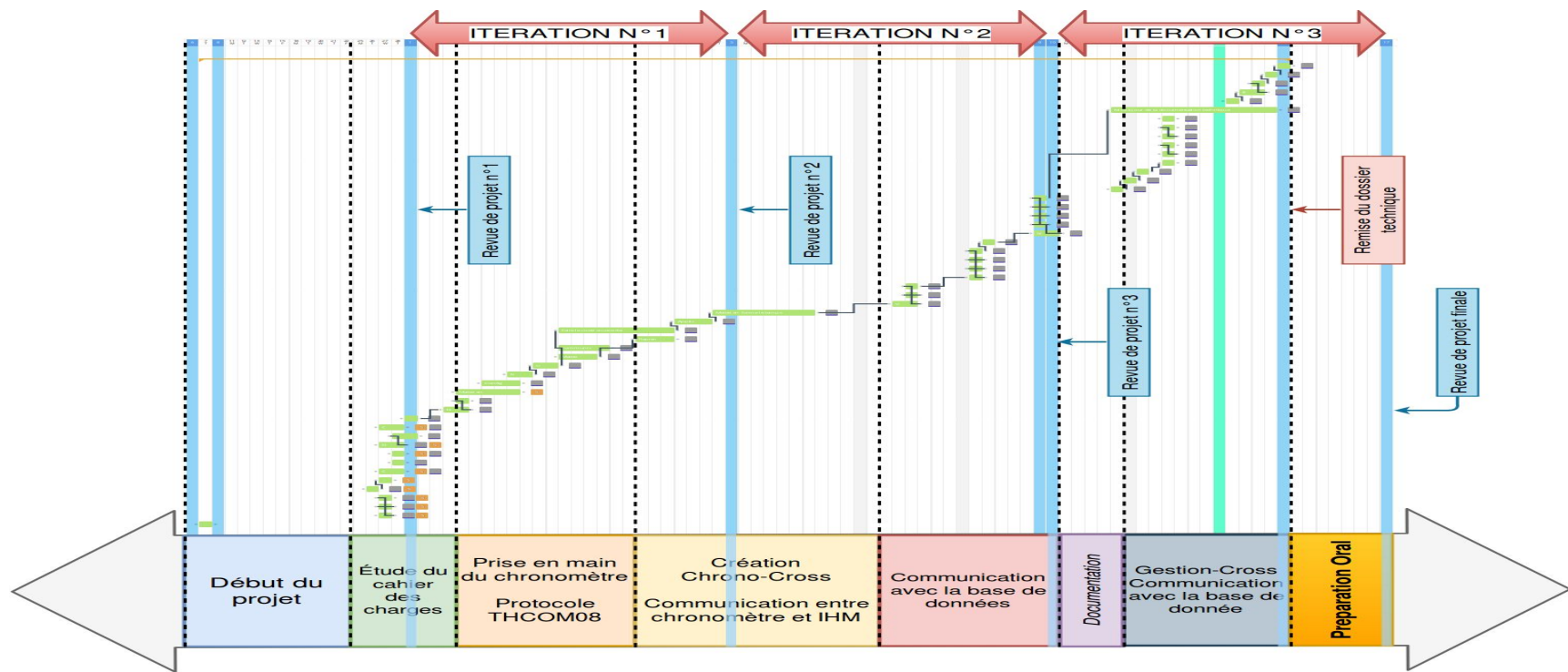
# Diagramme des cas d'utilisation



# Planification

Cas d'utilisation	Priorité	Itération
Chronométrer et classer les arrivées	Haute	1
Inscrire les coureurs et leur attribuer un dossard	Moyenne	2
Éditer les coureurs	Basse	3

# Planification



# Les ressources de développement

Interface de programmation :

- Qt 5.11.2



Langage de programmation :

- C++



Environnement de développement :

- QtCreator 4.7.1

Système de Gestion de Base de Données :

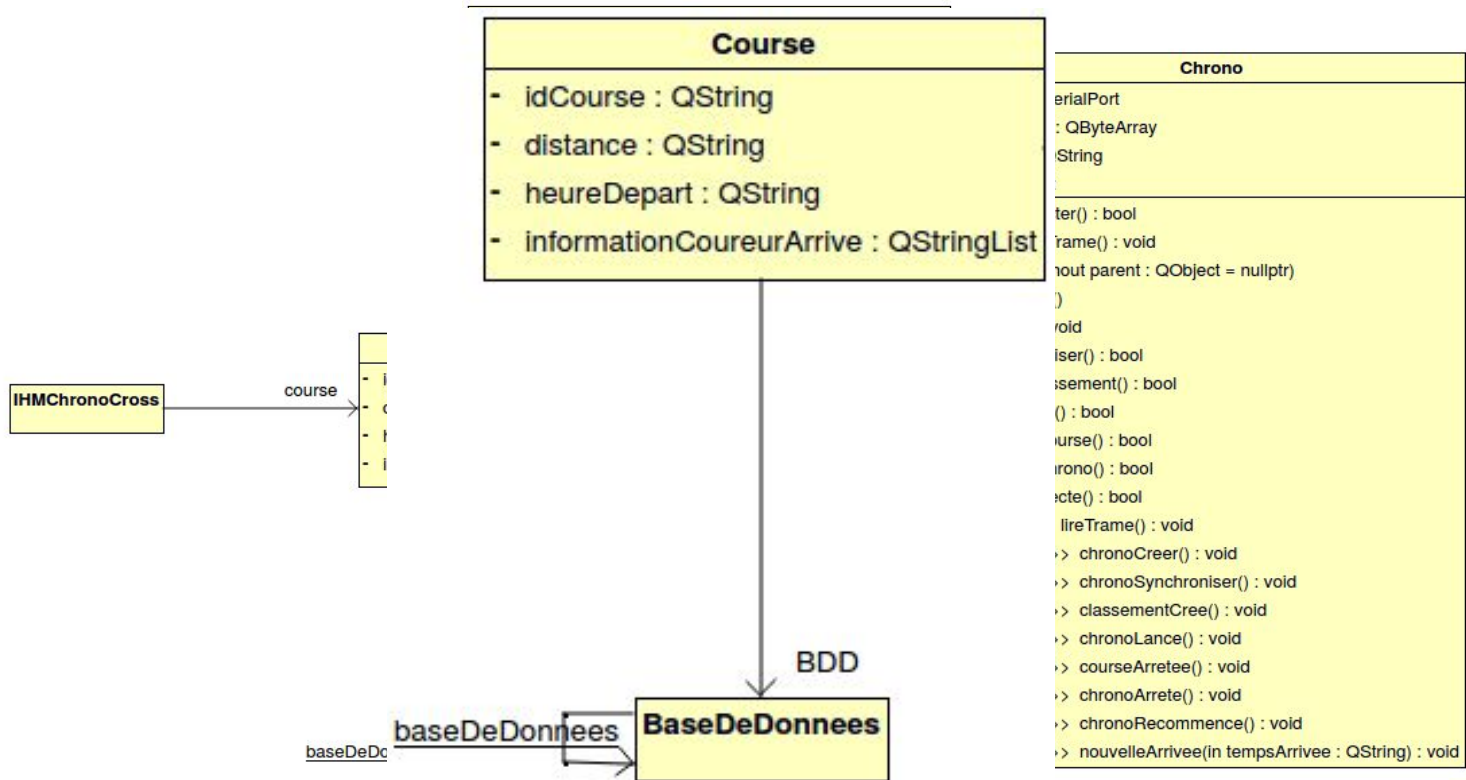
- Serveur MySQL



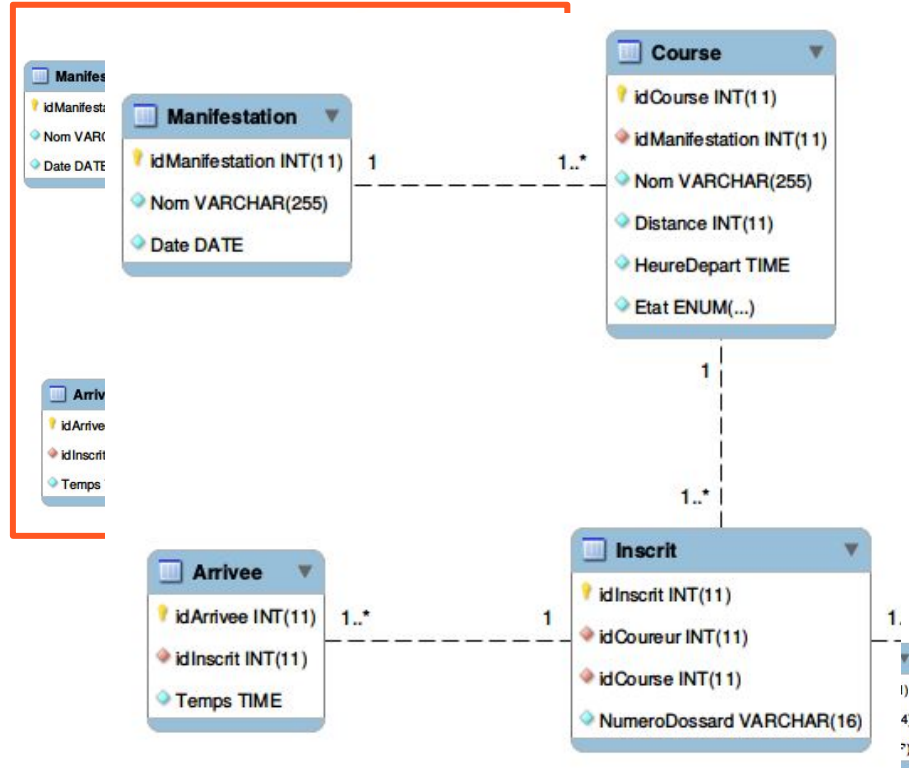
# Présentation personnelle



# Diagramme de classes (Chrono-Cross)

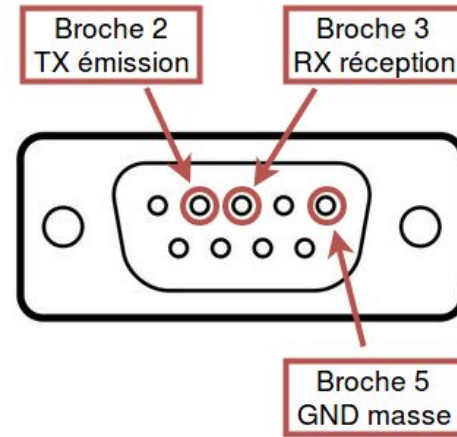


# Diagramme de base de données - CC



# Communication

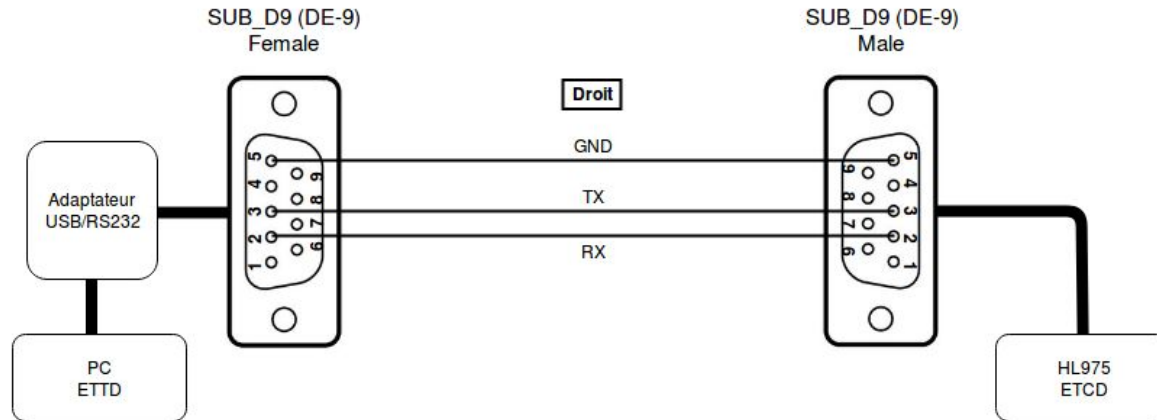
- 1 câble avec connecteur SUB\_D9 et 1 adaptateur RS232 / USB
- Liaison point à point - protocole RS232 (V24 EU)
- Liaison asynchrone
- Brochage
  - broche 2 - TX émission
  - broche 3 - RX réception
  - broche 5 - GND masse



# Communication

Cablage droit :

ETTD (DTE US): Équipement terminal de traitement de données  
ETCD (DCE US) : équipement terminal de circuit de données



# Configuration du port RS232

- Débit à 9600 bps ;
- 1 bit de start ;
- 8 bits de données ;
- Pas de parité ;
- Pas de control de flux ;
- 1 bit de stop ;

niveau logique	Polarité	Intervalle	Valeur typique
1	Basse	-3 V à -15 V	-12 V
0	Haute	3 V à 15 V	12 V

# Protocole THCOM08

## Trame utilisées:

### - Trame CLOSERUN :

Permet de mettre fin à la course pour le chrono (classement)

#	WC_001	\t	014B	\r	\n
---	--------	----	------	----	----

*On peut émettre deux styles de trame:*

*#WP Écrire un paramètre*

*#WC Écrire une commande*

---

### - Trame STARTLIGHT :

Permet de déclarer une nouvelle course avec son chrono et le style de départ

#	WP_120_3	\t	01AD	\r	\n
---	----------	----	------	----	----

# Trames d'acquiescement

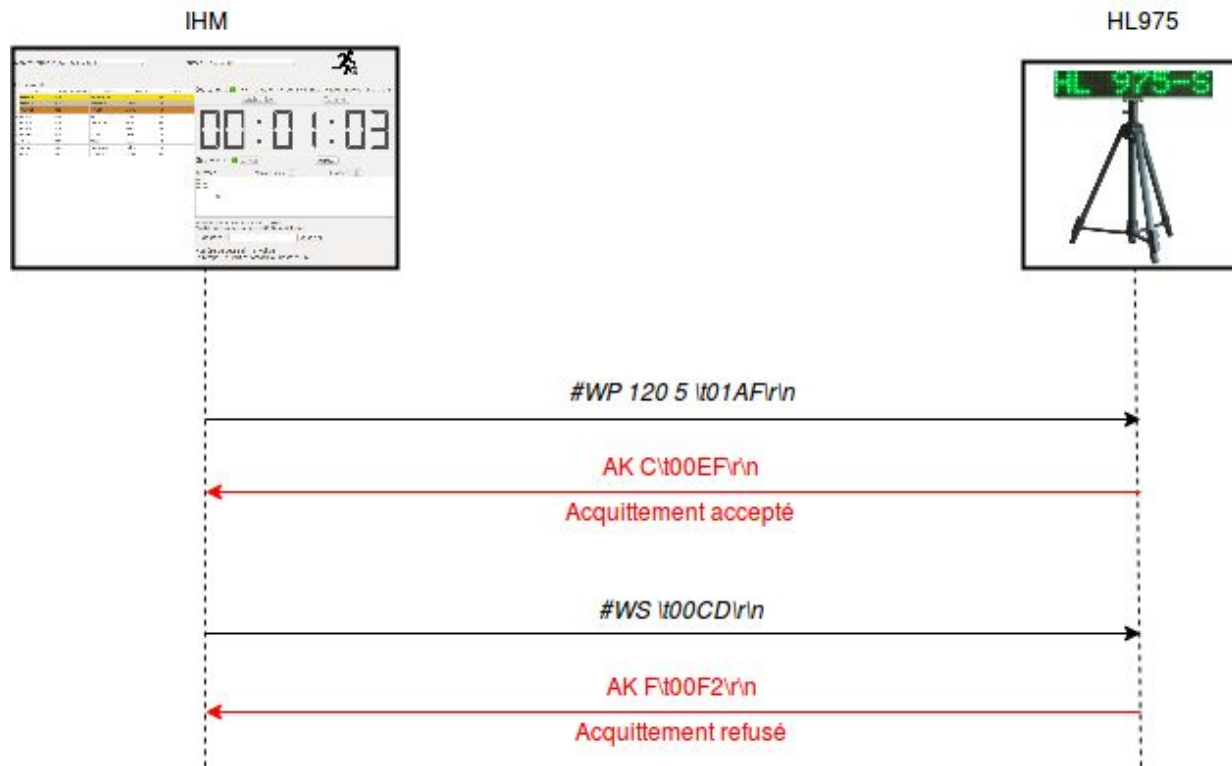
Se présente sous la forme :

AK_X	TAB	CS16	CR	Lf
------	-----	------	----	----

où X peut être

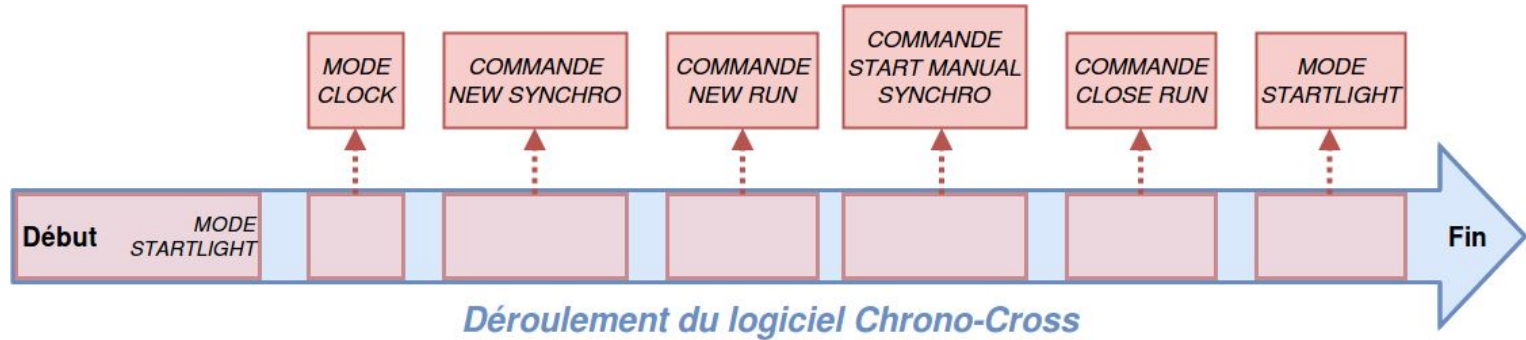
- **C** pour accepté : “AK\_C\t00EF\r\n”
- **F** pour rejeté : “AK\_F\t00F2\r\n”
- **R** pour non supporté : “AK\_R\t00FE\r\n”

# Trames d'acquittement





# Déroulement du logiciel Chrono-Cross



## Méthode decoderTrame()

```
void Chrono::decoderTrame()
{
    trame.remove("\r\n");
    if(trame.startsWith(TRAPE_AQUITTEMENT))
    {
        champsTrame = trame.split(" ", QString::SkipEmptyParts);
        if(champsTrame[1].startsWith('C'))
        {
            if(etat == ETAT_NONSYNCHRO)
            {
                setEtat(ETAT_MODECLOCK);
                emit chronoCreer();
            } else if(etat == ETAT_MODECLOCK)
            {
                setEtat(ETAT_NEWSYNCHRO);
                emit chronoSynchroniser();
            } else if(etat == ETAT_NEWSYNCHRO)
            {
                setEtat(ETAT_NEURUN);
                emit classementCreer();
            }
            else if(etat == ETAT_NEURUN)
            {
                setEtat(ETAT_STARTMANUALSYNCHRO);
                emit cronolance();
            }
            else if(etat == ETAT_STARTMANUALSYNCHRO)
            {
                setEtat(ETAT_CLOSUREN);
                emit courseArretee();
            }
            else if(etat == ETAT_CLOSUREN)
            {
                setEtat(ETAT_MODEND0);
                emit chroAnaArrete();
            }
            else if(etat == ETAT_MODEND0)
            {
                setEtat(ETAT_NONSYNCHRO);
                emit chronoRecommence();
            }
        }
    }
}

else if(trame.startsWith(TRAPE_TEMPSS))
{
    champsTrame = trame.split(" ", QString::SkipEmptyParts);
    emit nouvelleArrivee(champsTrame[champs_Traine_Temps]);
}

// Etat de l'etat == ETAT_NONSYNCHRO
{
    this->reconnecter();
}
```

```

if(champsTrame[1].startsWith("C"))
{
    if(etat == ETAT_NONSYNCHRO)
    {
        setEtat(ETAT_MODECLOCK);
        emit chronoCreer();
    }
    else if(etat == ETAT_MODECLOCK)
    {
        // etat c
        #define T
        #define T
        #define T
        // etat c
        #define E
        #define E
        #define E
        #define E
        #define E
        #define E
        #define E
    }
}

void Chrono::decoderTrame()
{
    trame.remove("\r\n");
    QStringList champsTrame;

    if(trame)
    {
        else if(etat == ETAT_NEWRUN)
        {
            setEtat(ETAT_STARTMANUALSYNCHRO);
        }
    }
    else if(etat == ETAT_NONSYNCHRO)
    {
        this->reconnecter();
    }

    {
        champsTrame = trame.split(" ", QString::SkipEmptyParts);
        emit nouvelleArrivee(champsTrame[CHAMPS_TRAJE_TEMP]);
    }

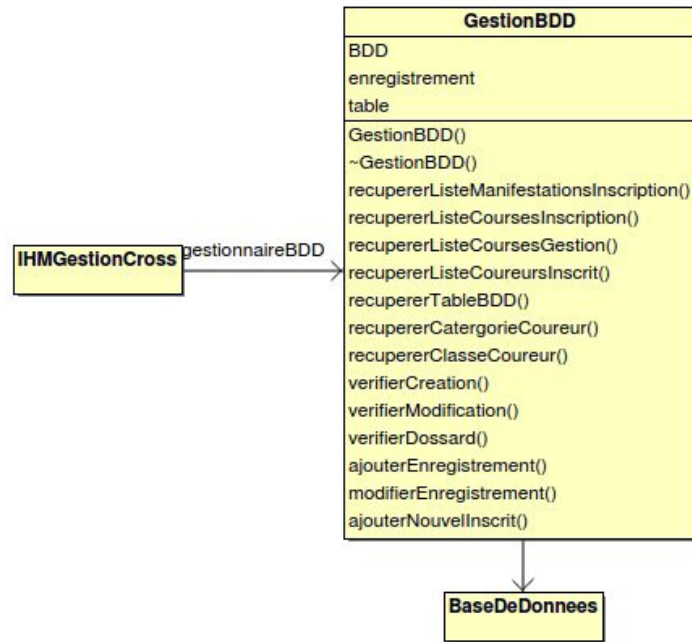
    }
    else if(etat == ETAT_MODEND)
    {
        setEtat(ETAT_NONSYNCHRO);
        emit chronoRecommence();
    }
}

```

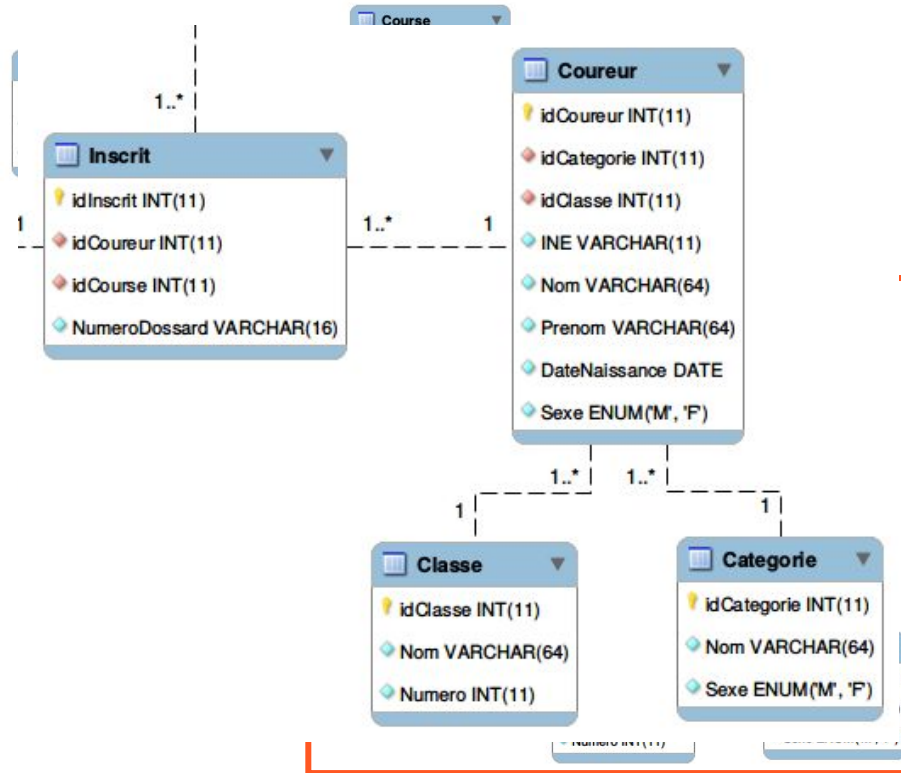
```
// PC <- TAG HEUER
#define TRAME_ACQUITTEMENT "AK"
#define TRAME_SYNCHRO "TS"
#define TRAME_PARAMETRES "&P"
#define TRAME_TEMPS "TN"
#define TRAME_COURSE_TERMINEE "CL"

// etat chrono
#define ETAT_NONSYNCHRO "NONSYNCHRO"
#define ETAT_MODECLOCK "MODECLOCK"
#define ETAT_NEWSYNCHRO "NEWSYNCHRO"
#define ETAT_NEWRUN "NEWRUN"
#define ETAT_STARTMANUALSYNCHRO "STARTMANUAL"
#define ETAT_CLOSERUN "CLOSERUN"
#define ETAT_MODEND "MODEND"
```

# Diagramme de classes (Gestion-Cross)



# Diagramme de base de données - GC

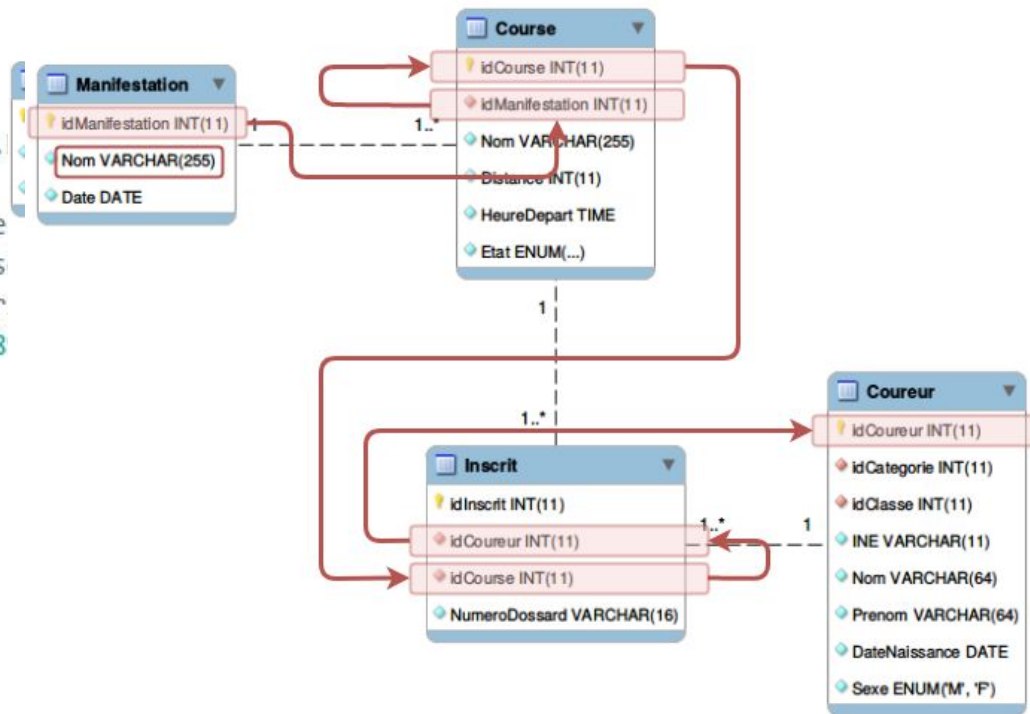


# Jointure

On souhaite récupérer le nom de:

```
SELECT Manifestation.Nom
FROM Manifestation
JOIN Course ON Manifestation.idManifestation=Course.idManifestation
JOIN Inscrit ON Course.idCourse=Inscrit.idCourse
JOIN Coureur ON Inscrit.idCoureur=Coureur.idCoureur
WHERE INE = '123456789AA'
```

```
SELECT Manifestation.
FROM Manifestation
JOIN Course ON Manife
JOIN Inscrit ON Cours
JOIN Coureur ON Inscr
WHERE INE = '12345678
```



# Les fonctionnalités & IHM :

## Itération 1 :

- Lancer et arrêter le chronomètre
- Synchroniser le chronomètre et l'IHM
- Signaler la connexion Chrono - IHM
- Recevoir les temps
- Attribuer les dossards au temps reçus

The wireframe shows a software interface for a timing application. At the top is a title bar labeled 'Départ' with a close button. Below it is a navigation bar with five tabs: 'Manifestations', 'Courses', 'Coureurs', 'Inscriptions', and 'Départ'. The 'Départ' tab is currently selected. The main content area contains a dropdown menu labeled 'Sélectionner la course' with a downward arrow. Below this is a table with 8 rows and 5 columns. The columns are labeled 'N° Dossard', 'Nom', 'Prenom', and 'Classe'. The rows are numbered 1 through 8. To the right of the table is a large black rectangular display showing the time in 'HH : MM : SS' format. Below the display are three buttons: 'Départ', 'Arrêt', and 'Imprimer'.


	N° Dossard	Nom	Prenom	Classe
1				
2				
3				
4				
5				
6				
7				
8				

*Maquette IHM*

# Les fonctionnalités & IHM :


It


Manifestations :  Courses :






Classement :

	Temps	Numéro de Dossard	Nom	Prenom	Classe
1	00:00:01	101	PERRICHON	Julia	4E
2	00:00:02	102	MOUTARD	Camille	4E
3	00:00:03	103	MOLIST	Lucille	4E
4	00:00:03	104	RIES	Clementine	4E
5	00:00:04	105	LAMOUREUX	Felicia	4E
6	00:00:08	106	STEY	Pauline	4E
7	00:00:09	108	BODIN	Alexia	4E
8	00:00:16	109	TERREC	Maele	4E
9	00:00:18	111	WINTREBERT	Pauline	4E
10	00:00:18	112	GOURLET	Romane	4E

Course Etat :  Nom : Cross M15 F Distance : 3500 Heure : 13:00:00 Inscrits : 15



Chrono Etat : 

Arrivées : Non classées:  Classées: 

00:00:18  
00:00:48  
00:00:49

Pour supprimer le premier temps non classées  
Veuillez entrer le numéro de dossard 0000 puis confirmer.

N° dossard :

Numéro de dossard 112 : validé  
Le temps 00:00:18 est associé au dossard 112

IHM CC  
version 1.0

dans le

oueur au

# Les fonctionnalités & IHM :

## Itération 3 :

- Créer, modifier ou supprimer un coureur
- Afficher les informations d'un coureur
- Inscrire et désinscrire un coureur à une course
- Attribuer un numéro de dossard à un coureur inscrit
- Afficher tous les coureurs enregistrés de la base de données

Manifestations
Courses
Coureurs

Nom	Prénom	Classe	Catégorie	INE	Date de Naissance	Sexe
PERRICHON	Julia	4E	M15	123456789AA	2004-04-16	F
MOUTARD	Camille	4E	M15	123456789BB	2004-01-08	F
MOLIST	Lucille	4E	M15	123456789CC	2004-09-26	F
LAMOUREUX	Felicia	4E	M15	123456789EE	2004-01-15	F
STEY	Pauline	4E	M15	123456789FF	2004-05-01	F
LIBERHESOUR	Marie	4E	M15	123456789GG	2004-01-21	F
BODIN	Alexia	4E	M15	123456789HH	2004-10-29	F
TERRE	Maëlle	4E	M15	123456789II	2004-11-05	F
FORNES	Marie	4E	M15	123456789JJ	2004-02-01	F
WINTREBERT	Pauline	4E	M15	123456789KK	2004-12-04	F
FORNES	Romane	4E	M15	123456789LL	2004-03-10	F
VINCENT	Ines	4E	M15	123456789MM	2004-04-01	F
DUTOT	Camille	4E	M15	123456789NN	2004-09-13	F
PREVOST	Emma	4E	M15	123456789OO	2004-06-04	F
DOE	Jane	4E	M15	123456789PP	2004-01-01	F
PELIOT	Julien	4E	M15	234567891AA	2004-02-16	M
MOUTARD	Charles	4E	M15	234567891BB	2004-03-08	M
DURAND	Lucien	4E	M15	234567891CC	2004-10-26	M
RIQUY	Camille	4E	M15	234567891DD	2004-07-07	M
AMOUR	Felicien	4E	M15	234567891EE	2004-08-24	M
STEY	Paul	4E	M15	234567891FF	2004-06-25	M
HERRAULT	Marine	4E	M15	234567891GG	2004-03-09	M
BOTIN	Alexis	4E	M15	234567891HH	2004-11-28	M
TORRES	Marcel	4E	M15	234567891II	2004-12-07	M
FURLES	Mario	4E	M15	234567891JJ	2004-08-01	M
INTERBERT	Alexandre	4E	M15	234567891KK	2004-07-01	M
GOLUSSET	Romain	4E	M15	234567891LL	2004-04-17	M

Coureur :

Nom : LAMOUREUX

Prénom : Felicia

Date de Naissance : 15/01/2004

Classe : 4E

Catégorie : M15

INE : 123456789EE

Sexe : ☒ F ☐ M

Participe à : Cross M15 F

Nouveau
Modifier
Supprimer

Inscription :

Manifestation : Cross Lycee 2019

Course : Cross M17 F

Numéro de dossard : 3

Inscrire

Nom	Prénom	Dossard
MARTIN	Emma	301
BERNARD	Louise	302
PETIT	Jade	303
BOYER	Alice	304
DUBOIS	Chloé	305
MOREAU	Lina	306

Pour inscrire un(e) coureur(euse)  
Veuillez entrer un numéro de dossard disponible entre 301 et 399

IHM GC  
version 1.0



# Les tests - Chrono Cross

Scénario	Résultat attendu	OUI / NON	Remarques
Démarrer une course	Envoie et réception de trame valide	OUI	
Afficher le chronomètre QLCD sur l'IHM	Le chronomètre de l'IHM est synchronisé avec le module	OUI	
Recevoir les temps	Lorsque qu'un coureur passe la ligne d'arrivée, le temps s'affiche	OUI	
Associer un dossard à un temps d'arrivée	Lorsque qu'un temps est reçu on peut lui attribuer un numéro de dossard et l'ajouter au classement	OUI	

# Les tests - Chrono Cross

Scénario	Résultat attendu	OUI / NON	Remarques
Mise en page particulière pour les trois premiers	Les trois premiers temps sont en texte gras de couleur : or, argent et bronze	OUI	
Supprimer le premiers temps non classée	Lorsque un temps non classés reçoit un dossard valide on l'ajoute à la table arrive	OUI	
Entrer un numéro de dossard erroné	Un message d'erreur apparaît pour prévenir l'organisateur	OUI	

# Les tests - Gestion Cross

Scénario	Résultat attendu	OUI / NON	Remarques
Creer un coureur	Ajoute un coureur à la base de données	OUI	
Modifier un coureur	Les informations du coureur ont été modifiés	OUI	
Supprimer un coureur	Le coureur est retiré de la base de donnée Coureur et de toutes les courses où il était inscrit	OUI	
Inscrire un coureur à une course en lui attribuant un dossard	Le numéro de dossard est ajouté à la liste d'inscrit pour la course	OUI	
Créer un coureur avec des informations erronées	Un message s'affiche pour préciser l'erreur	OUI	

# Améliorations possibles :



- Augmenter la précision du chronomètre (si nécessaire)
- Utiliser les autres fonctionnalités (countdown, light, text)

- Utiliser un pistolet de départ pour lancer la course



- Utiliser un faisceau infra-rouge pour lancer la course lorsqu'il est franchi (avec un compte-à-rebours)

**Je vous remercie de votre attention**

Michaël ANDREO