

# Projet Praxis

## *Retour des tests mis en place par le Cati Sicpa*

### Introduction

Ce document ne traite que des retours de tests menés par le Cati Sicpa. Les résultats des tests menés dans le cadre du logiciel Adonis sont reportés dans un autre document.

### Résumé du projet

Etudier les solutions, matériels et logiciels, pour équiper les unités expérimentales animales et végétales d'INRAE pour la saisie des données terrain en remplacement du matériel existant devenu obsolète. La contrainte étant de garder les atouts actuels : Appareils de terrain mobiles, robustes, équipés d'antenne RFID (Radio Frequency Identification) et/ou de lecteur de codes-barres qui permettent le développement de logiciels métiers.

### Demande de financement

Une demande de financement a été déposée en avril 2020 auprès de la DipSO. Le budget obtenu a permis d'acheter des appareils mobiles robustes sous Android, du matériel RFID Bluetooth et financer une formation au framework Xamarin. Cette demande a été portée par le Cati Sicpa en collaboration avec le projet Adonis.

### Contexte du projet

Les Unités Expérimentales animales et végétales d'INRAE utilisent, pour la saisie des données terrains, des appareils mobiles durcis équipés de différents types de lecteurs : RFID, codes à barre 1D ou 2D (Ex : Workabout Pro 3 de chez Psion Zebra). Dans les UE animales, les terminaux mobiles durcis sont utilisés pour l'enregistrement des données expérimentales et d'élevages (naissance, pesées, traitement sanitaire, ...) pour les systèmes d'informations du Cati Sicpa « espèces » (bovins, porcins, volailles, petits ruminants, lapins, poissons) et « multi-espèces » (sanitaire et alimentation). La robustesse des appareils permet de les utiliser dans les conditions d'élevage : proche des animaux, risque de chute et donc de piétinement animal, ... De plus, l'antenne RFID permet de lire automatiquement la boucle électronique des animaux et ainsi enregistrer de manière fiable le numéro de l'animal ou le numéro de bassin. La capacité de la batterie permet d'effectuer des chantiers de saisie de données de plusieurs heures sans remettre en charge le terminal mobile.

Actuellement, le système d'exploitation Windows Mobile, permet aux agents du Cati Sicpa de développer des logiciels spécifiques liés aux besoins des utilisateurs. Ces développements se font en C# en utilisant le compact Framework .NET 3.5. Une soixantaine de terminaux mobiles sont utilisés dans les UE animales avec une vingtaine d'applications déployées.

Dans le domaine du végétal, c'est le logiciel Adonis, développé aussi en C# pour Windows Mobile, qui permet l'acquisition sur le terrain des données d'expérimentations.

Microsoft abandonnant son OS Windows Mobile, notre fournisseur a annoncé l'arrêt de la commercialisation de l'appareil que nous utilisons. Les tests menés dans le cadre de ce projet visaient donc à tester des terminaux mobiles sous un OS Android, mais, équivalent à l'appareil précédent dans ces caractéristiques techniques. Le volet développement d'applications sous Android avec l'utilisation du framework Xamarin a aussi été évalué dans ce contexte.

Inra - Cati Sicpa Systèmes d'Informations et Calcul pour le Phénotypage Animal	Projet Praxis	Code : Praxis_RetourTests
	Retour des tests	Date : 31/03/2021 Rédacteur(s) : A. Journaux / J-F. Bompa

## Tests des appareils

### Inventaire des appareils achetés

Le choix d'acheter et de tester 3 terminaux mobiles sous Android avec des caractéristiques différentes a été fait, pour essayer de déterminer la "meilleure expérience utilisateur".

- 1 terminal IP65 avec clavier et écran 4.3" (Cipherlab RK95)
- 1 terminal IP65 plus compact avec clavier et écran 4" (Handheld Nautiz X41)
- 1 terminal IP65 sans clavier avec écran 4.3" et lecteur RFID basse fréquence (Coppernic C-One<sup>2</sup>)

Les 3 terminaux sont équipés d'un lecteur code barre, d'une caméra, du BT et Wifi, et d'un socle de bureau.

Etant donné que l'utilisation d'un lecteur RFID est indispensable pour lire les tags de nos animaux d'élevage, deux lecteurs RFID portables indépendants ont aussi été achetés.

- 1 lecteur base fréquence Bluetooth (Agrident APR 600)
- 1 lecteur basse fréquence en kit (Agrident module ABR200) pour essayer de l'intégrer sur un des terminaux mobiles

### Caractéristiques techniques testés

Les caractéristiques générales des appareils conviennent pour notre utilisation : mémoire, puissance processeur, luminosité d'écran, Wifi, Bluetooth et indice de protection.

Une attention plus particulière a été portée sur l'autonomie, les performances des lecteurs code barre et RFID pour le C-One.

#### Les lecteurs de codes-barres

Les lecteurs sont tous très performants.

- Bonne distance de lecture, 20-30 cm
- Lecture de très petits codes-barres 1D et 2D
- Intégration d'une émulation clavier

#### Lecteur RFID

Le lecteur RFID RF Agrident est suffisamment performant pour nos besoins, distance de lecture entre 10 et 20 cm selon les tags. Mêmes caractéristiques que celui du Psion WAP4. Coppernic a aussi intégré une émulation clavier.

#### Autonomie

Test de l'autonomie avec le Wifi connecté, un lecteur code-barre connecté en Bluetooth, la mise en veille à 30mn et l'écran en luminosité maxi.

	CiperLAB RK95	NAUTIZ X41	Coppernic C-ONE <sup>2</sup>
	Batterie		
9h00	100%	100%	100%
10h18	85%	95%	92%
11h00	76%	91%	87%
12h00	63%	84%	80%
12h56	51%	79%	73%
14h00	37%	72%	65%
15h00	22%	66%	57%
16h00		59%	49%
17h00		53%	42%

On voit que l'on peut facilement tenir au moins 6h pour le RK95 et plus de 8h pour les autres.

Le moins bon résultat obtenu par le RK95 est dû à son écran beaucoup plus lumineux. Quand on règle la luminosité à 75%, on retrouve à peu-près la même autonomie que les 2 autres.

Inra - Cati Sicpa Systèmes d'Informations et Calcul pour le Phénotypage Animal	Projet Praxis	Code : Praxis_RetourTests
	Retour des tests	Date : 31/03/2021 Rédacteur(s) : A. Journaux / J-F. Bompa

## Les lecteurs RFID Bluetooth

La solution de lecture RFID pour le X41 et le RK95 passe obligatoirement par la connexion Bluetooth à un lecteur portable indépendant. L'APR 600 d'Agrident est un des lecteurs RFID portable performant et facile d'utilisation avec un bel écran couleur. Il est sur batterie avec une très bonne autonomie. Il possède des options de vibreur, paramétrage des trames envoyées, un clavier numérique et la possibilité d'enregistrer des listes de façon autonome.

L'autre solution "en kit" utilise le même module RFID mais il faut refaire un lecteur simplifié. L'idée est de le rendre plus compact en enlevant l'écran et le clavier. Il n'y a pas besoin d'option. Le simple fait de renvoyer le numéro RFID par Bluetooth sera suffisant. Pour l'instant, cette partie là n'a pas encore pu être testée.

## Bilan

Le bilan des tests des 3 terminaux mobiles reste très satisfaisant. Les caractéristiques générales conviennent et les équipements sont de bonne qualité. Les sensations de tailles, poids et prises en main sont meilleures qu'avec le WAP4. Les touches du clavier du RK95 sont un peu petites mais restent exploitables. En première impression, le manque du clavier physique sur le C-one<sup>2</sup> ne semble pas être une contrainte. On arrive à l'utiliser avec les doigts mouillés et même avec des gants en plastiques fins. De plus un stylet est fourni pour plus de précision tactile ou pour l'utilisation avec des gants épais. Les 3 appareils nous permettent de proposer 3 expériences utilisateurs un peu différentes.

**RK 95 :** grande taille, grand écran, grand clavier

**X41 :** taille compacte, petit écran et petit clavier

**C-ONE<sup>2</sup> :** taille compacte, grand écran, lecteur RFID

Bien sur notre préférence tend vers le C-ONE<sup>2</sup> car c'est le seul à intégrer un lecteur RFID performant, et donc l'utilisateur n'a besoin que d'une seule main.

<b>Inra - Cati Sicpa</b> Systèmes d'Informations et Calcul pour le Phénotypage Animal	Projet Praxis	Code : Praxis_RetourTests
	Retour des tests	Date : 31/03/2021 Rédacteur(s) : A. Journaux / J-F. Bompa

## Test développement logiciel

### Compétences en interne

Les applications utilisées dans le cadre du phénotypage animal sont développées en interne par des agents du Cati Sicpa. Que ce soit pour des applications client/serveur ou des applications mobiles, ces développements sont réalisés avec le langage C#. Une compétence interne et une communauté active existent donc, pour ce langage, au sein du Cati Sicpa.

Notre souhait d'utiliser une technologie qui valorise cette compétence interne, nous a conduit à s'intéresser au framework Xamarin.

### Framework Xamarin

Xamarin est un ensemble de bibliothèques open source qui permet d'écrire des interfaces utilisateurs totalement portable sous toutes les plateformes mobiles mais pas seulement puisque le code peut tourner sur Mac et bien sûr sur PC... Xamarin est une couche d'abstraction qui gère la communication entre le code partagé et le code de la plateforme cible. Cela permet aux développeurs de partager en moyenne 90 % de leur application entre les plateformes. Les développeurs peuvent ainsi écrire toute leur logique métier dans un seul langage (ou de réutiliser du code d'application existant) tout en gardant une interface native sur chaque plateforme.

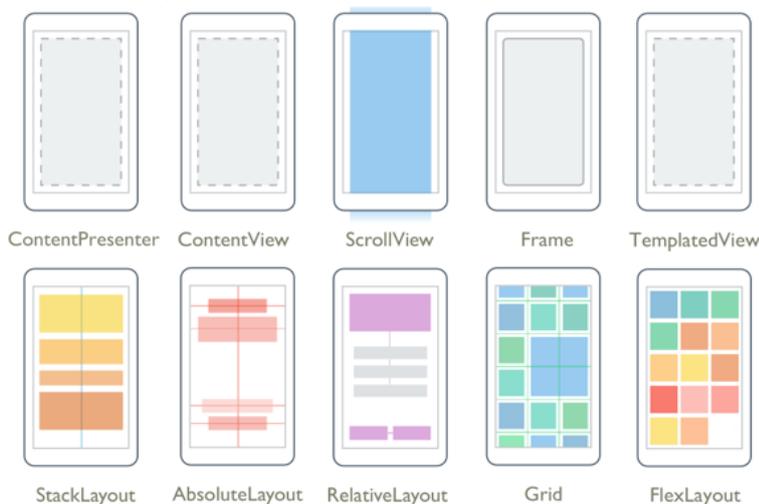


L'avantage principal, pour nous, est donc, de continuer à développer en C# pour déployer sur des plateformes Android ; et ainsi, partager toute la logique métier déjà existante.

### Xamarin.Forms

Xamarin.Forms permet aux développeurs de créer des interfaces utilisateur en XAML partagé quel que soit la plateforme cible.

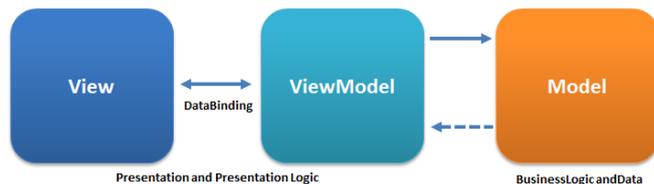
Xaml est un langage similaire à xml basé sur le principe de balise. Il permet de dessiner facilement les interfaces grâce à un ensemble de Layout.



<b>Inra - Cati Sicpa</b> Systèmes d'Informations et Calcul pour le Phénotypage Animal	Projet Praxis	Code : Praxis_RetourTests
	Retour des tests	Date : 31/03/2021 Rédacteur(s) : A. Journaux / J-F. Bompas

## Pattern MVVM

Développer en Xamarin, c'est développer en utilisant le patron d'architecture MVVM (Model-View-ViewModel). Il permet de séparer les données (Model), la logique (ViewModel) et l'interface utilisateur (View).



- La View est décrite en XAML de façon purement déclarative.
- Le Model est la représentation d'une donnée, au sens métier du terme. Les classes qui décrivent le modèle peuvent donc provenir d'un autre projet qui utiliserait la même couche métier.
- Le ViewModel est la couche intermédiaire qui prépare les données et les expose à la View. C'est généralement là qu'on trouvera la logique de validation des données, l'exécution des calculs, l'appel aux différents services...

## Xamarin.Essentiels

Xamarin.Essentials est une bibliothèque qui fournit des API multiplateformes pour faciliter l'accès aux fonctionnalités natives des appareils. Voici quelques exemples de fonctionnalités fournies par Xamarin.Essentials :

- Accéléromètre
- Niveau de la batterie
- Géolocalisation
- Informations sur l'appareil
- ...

## Apprentissage de Xamarin

Une formation de 5 jours a été suivie par 6 agents du Cati Sicpa en mars et avril 2021. Cette formation requière des solides compétences en programmation objet sous C#. Une première marche est à franchir avec l'apprentissage du Xaml. Mais la marche la plus délicate est l'apprentissage de tout le mécanisme MVVM.

Suite à cette formation, afin d'asseoir et de partager un socle de connaissances sur cette technologie, un support de cours « Bien démarrer avec Xamarin » a été rédigé. Il est accessible à cette adresse :

<https://forge-dga.jouy.inra.fr/attachments/download/12255/BienDemarrerAvecXamarin.pdf>