



Conception et Réalisation d'un prototype afficheur des horaires de prières

Réalisé par

MANI Mohamed Anis

Encadré par

Mr BESKRI Charfeddine

2011/2012

Plan

- Problématique
- Étude du besoin
- Cahier de charges
- Partie Matérielle
- Partie Logicielle
- Statistiques
- Conclusion

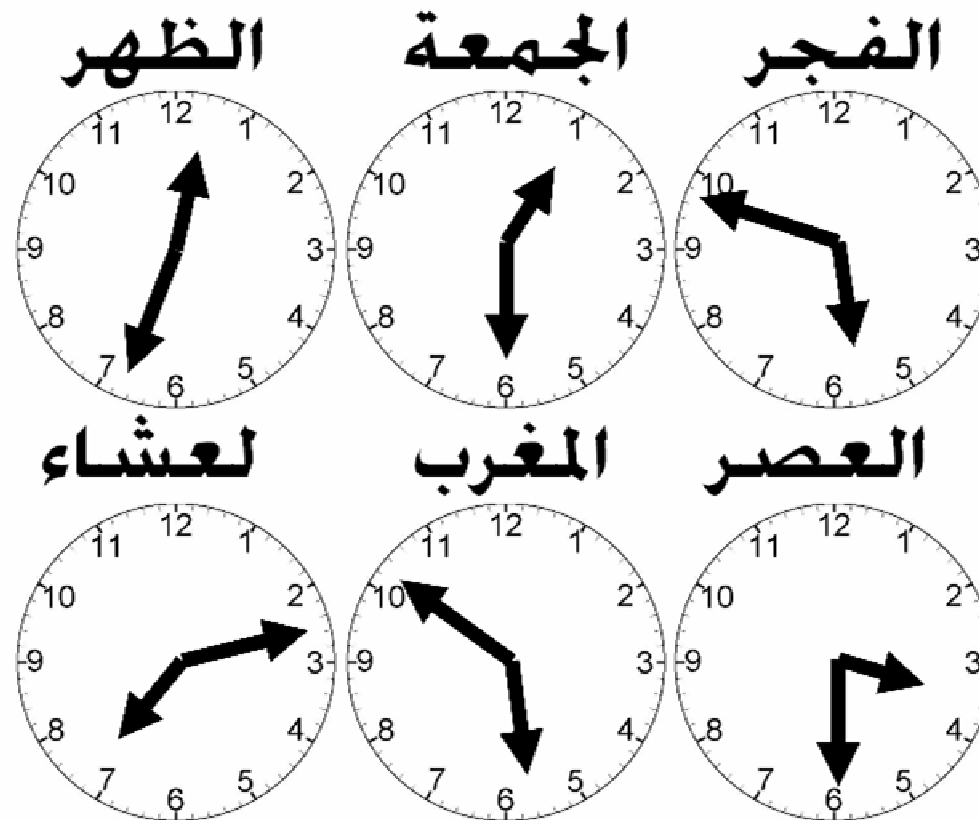
Problématique (1 / 3)

- Importance de faire la prière à son temps et dans les locaux destinés pour ce rituel
 - **Non valable** : si faite avant l'arrivée de son temps
 - **Dette** : si faite après la sortie du temps
 - **Mieux récompensée** : si faite avec l'« imam » dans la mosquée
- ➔ Besoin de connaître les horaires de prière, pour être avec l'« imam » et bénéficier de la récompense.

Problématique (2/3)

- Après la révolution tunisienne, le citoyen peut fréquenter librement les mosquées sans contraintes et sans poursuites par la police politique.
 - Le nombre de pratiquants s'est multiplié
 - Beaucoup de nouvelles mosquées sont en construction
 - Chaque mosquée doit afficher les horaires de prières diffusées par le Ministère des affaires religieuses
 - L'affichage des horaires de prières est réalisé sur un tableau statique mis à jour par le responsable de la mosquée
 - La non mise à jour de ces horaires risque d'induire en erreur les pratiquants
- ➔ Besoin de concevoir un système automatique qui permet d'afficher les horaires de prières.

[Problématique (3/3)]



Exemple d'affichage statique des horaires de prières

Étude du besoin

- La société FOUNOUN EL KANTAOUI, est une société qui agit dans le secteur de l'affichage publicitaire statique. **Elle offre les tableaux d'affichages statiques des horaires de prières à des prix compétitifs.**
- La société souhaite se lancer dans le secteur de l'affichage dynamique/numérique en commercialisant les tableaux **d'affichage numérique des horaires de prières vu l'augmentation de la demande sur ces produits dans la période actuelle.**
- La réalisation d'un prototype permettra à cette société de :
 - Évaluer les coûts de réalisation
 - Connaître les risques à encourir en finançant ce projet.
- ➔ **Réaliser un prototype d'un montage pour afficher les horaires de prières « Prayer-Caller »**

Cahier de charges (1 / 2)

- Modèle FURPS (Functionnality, Usability, Reliability, Performance & Supportabilité)
- **Exigences fonctionnelles**
 - Afficher la date/l'heure et les horaires des prières
 - Diffuser l'Azan aux horaires de prières
- **Exigences non fonctionnelles**
 - **Utilisabilité**
 - Usage simple sans apprentissage préalable
 - Accessible à distance à l'aide de télécommande
 - Affichage clair et visible de loin
 - **Fiabilité**
 - Les horaires doivent être suffisamment proches de la réalité
 - Fonctionner sans surchauffe, ni défaillances dans une plage de tension
 - Fonctionner sous batterie 7,5V à 24V / panneau photovoltaïque

Cahier de charges (2/2)

- Modèle FURPS (Functionnality, Usability, Reliability, Performance & Supportabilité)
- **Exigences non fonctionnelles**
 - **Performances**
 - Sauvegarde des paramètres du système même hors tension
 - **Maintenabilité**
 - Modulaire afin de cerner les emplacement de la défaillance
 - Utiliser des composants non onéreux et disponibles sur le marché



Partie Matérielle

Plan - Partie Matérielle

- Conception Modulaire
 - Procédure
 - Outils
- Système sonore
- Affichage des horaires de prière
- Interface de commande
 - Afficheur LCD Série
 - Commande Infrarouge
- Horaires de prières
 - Horloge RTC
 - Mémoire Série

Partie Matérielle

Conception Modulaire (1 / 2)

- Procédure
 - Vue d'ensemble
 - Conception/Réalisation
 - Conception
 - Choisir les composants
 - Consulter la documentation (datasheet) des composants
 - Dessiner le schéma de principe
 - Simulation
 - Implémentation & Réalisation
 - Routage
 - Réalisation du circuit imprimé
 - Soudure des composants
 - Programmation des μC
 - Tests Unitaires
 - Assemblage des modules
 - Programmation du module principal

Partie Matérielle

Conception Modulaire (2/2)

■ Outils

○ ISIS Proteus

- Dessiner le schéma de principe
- Réaliser les simulations

○ ISIS ARES

- Effectue le routage

○ PIC C

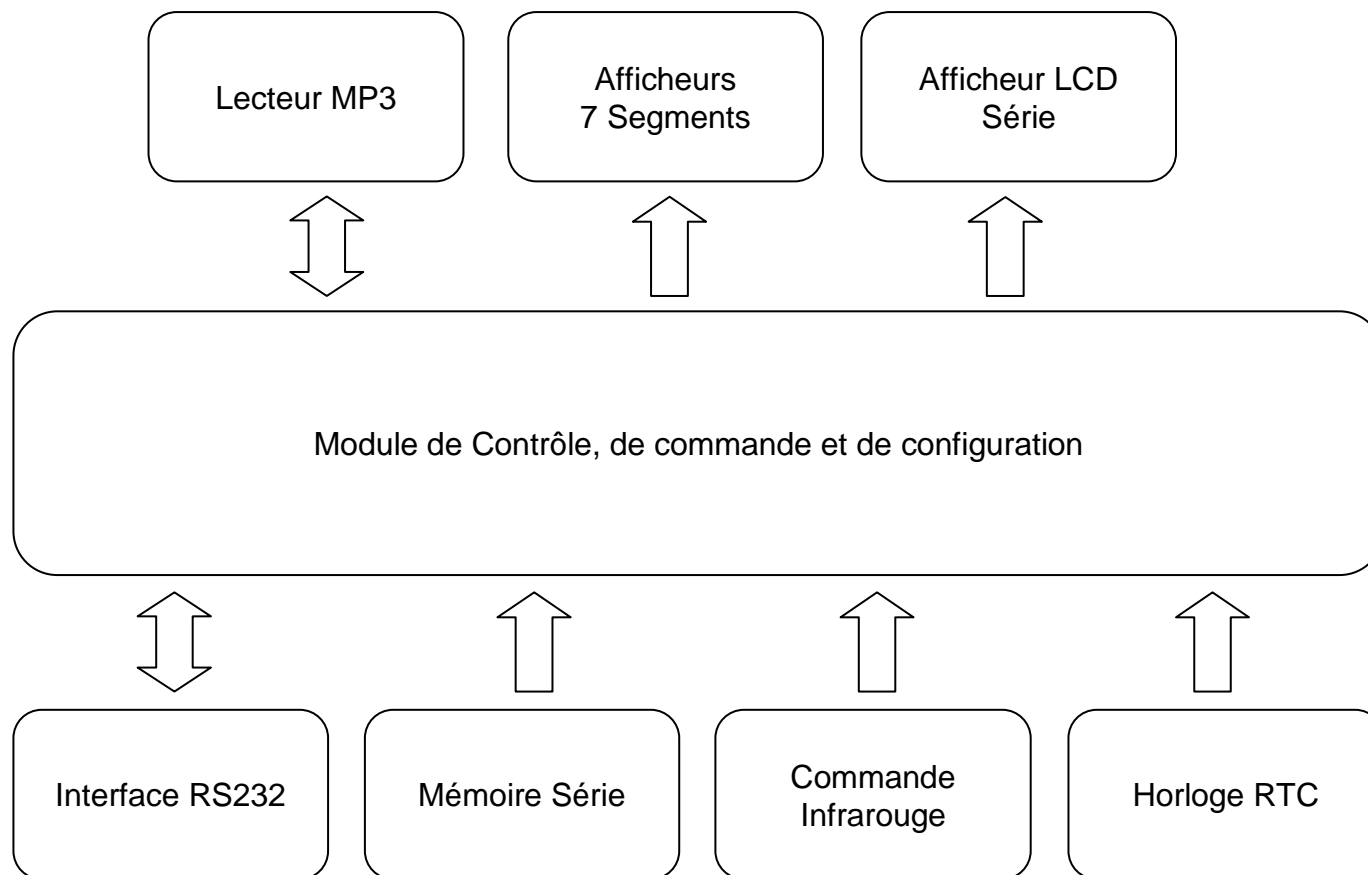
- Développement des programmes du μC

○ WINPIC

- Upload des programmes dans le μC

Partie Matérielle

Vue d'ensemble Globale



Partie Matérielle

Système Sonore (1 / 4)

- **Fonction implémentée**
 - Diffuser l'appel à la prière
- **Solution utilisée**
 - Adapter un lecteur MP3 vendu dans le commerce à prix avantageux pour notre système
- **Contraintes**
 - Réaliser le mécanisme de contrôle
 - Émuler le système de commande



Figure 5, Interfaces de contrôle et de commande du lecteur MP3

Partie Matérielle

Système sonore (2/4)

■ Interface de contrôle

○ But

- Connaître l'état du lecteur MP3
 - Détecter le début et la fin de lecture
 - Connaître la position actuelle de lecture

○ Procédure

- Récupérer les signaux présents sur l'afficheur du lecteur
- Décoder ces signaux pour en déduire l'état du lecteur

■ Interface de commande

○ But

- Déclencher le début et la fin de lecteur pour jouer l'Azan

○ Procédure

- Dessiner le chronogramme du système de commande (Capteur Infrarouge)
- Envoi d'une trame de commande au capteur Infrarouge du lecteur

J



Partie Matérielle

Système sonore (4/4)

■ Vue d'ensemble

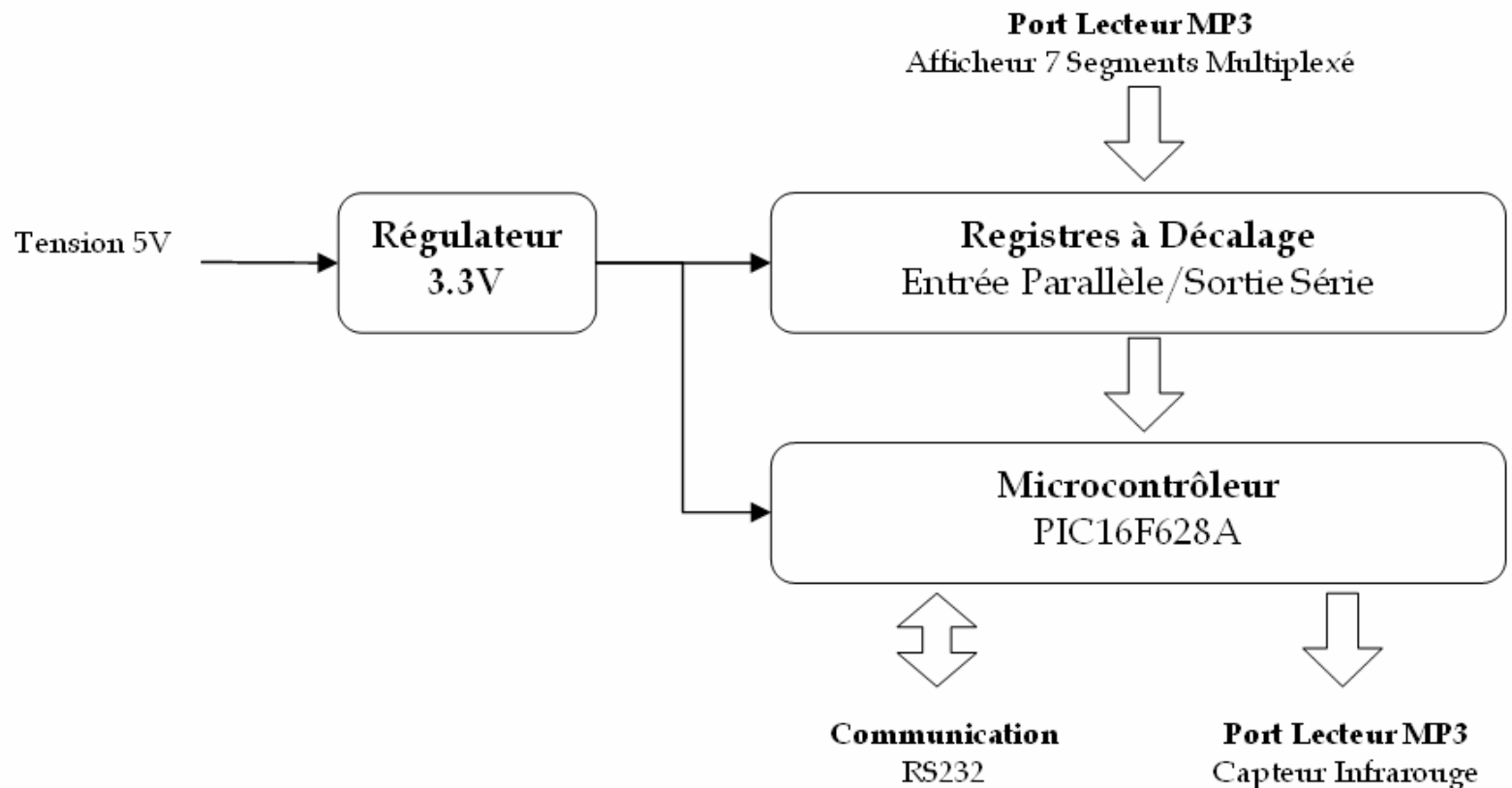


Figure 9, Vue d'ensemble de l'interface de commande du lecteur MP3

Partie Matérielle

Affichage des horaires de Prières (1 / 2)

■ Fonction implémentée

- Afficher des horaires de prières (tout en respectant les exigences énoncés dans le cahier de charges)

■ Solution utilisée

- Utiliser des afficheurs 7 segments (4 digits) multiplexé

■ Contraintes

- Comprendre le fonctionnement multiplexé
- Calculer le courant de commande de l'afficheur
- Effectuer un affichage stable
- Afficher une chaîne de caractères de longueur quelconque (inférieure à 32 caractères) en utilisant l'afficheur à 4 digits
- Choisir un protocole de commande pour l'afficheur

Partie Matérielle

Affichage des horaires de Prières (2/2)

■ Vue d'ensemble

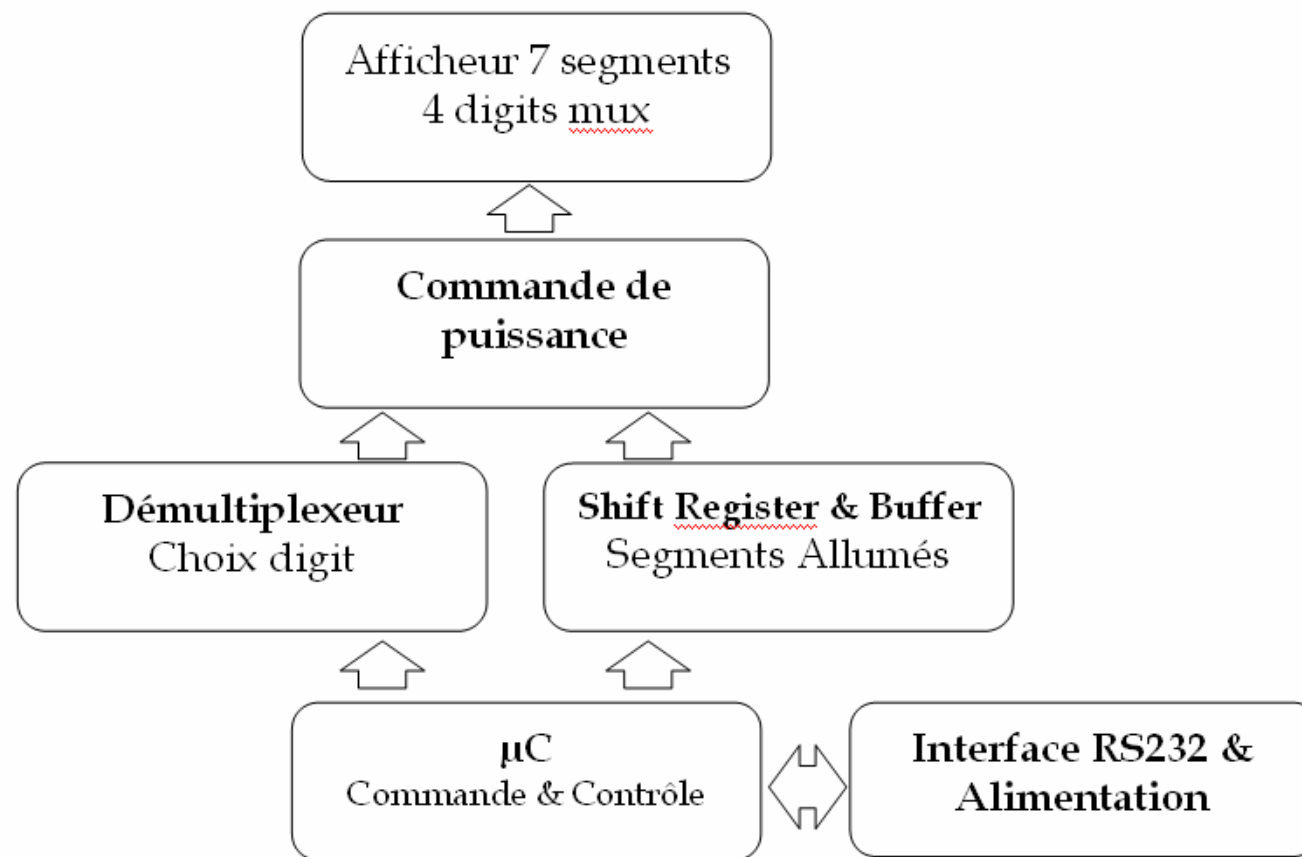


Figure 14, Vue d'ensemble de l'afficheur sept segments

Partie Matérielle

Interface de commande (1/3)

■ Fonction implémentée

- Permettre à l'utilisateur de contrôler/configurer le système

■ Solution utilisée

- Utiliser un afficheur LCD pour afficher les consignes de configuration
- Utiliser une télécommande pour commander le système

■ Contraintes

- Réduire le nombre de signaux de commande de l'afficheur LCD parce qu'il doit se situer loin du système de commande.
- Décrypter les signaux issues par la télécommande.

Partie Matérielle

Afficheur LCD Série (2/3)

■ Vue d'ensemble

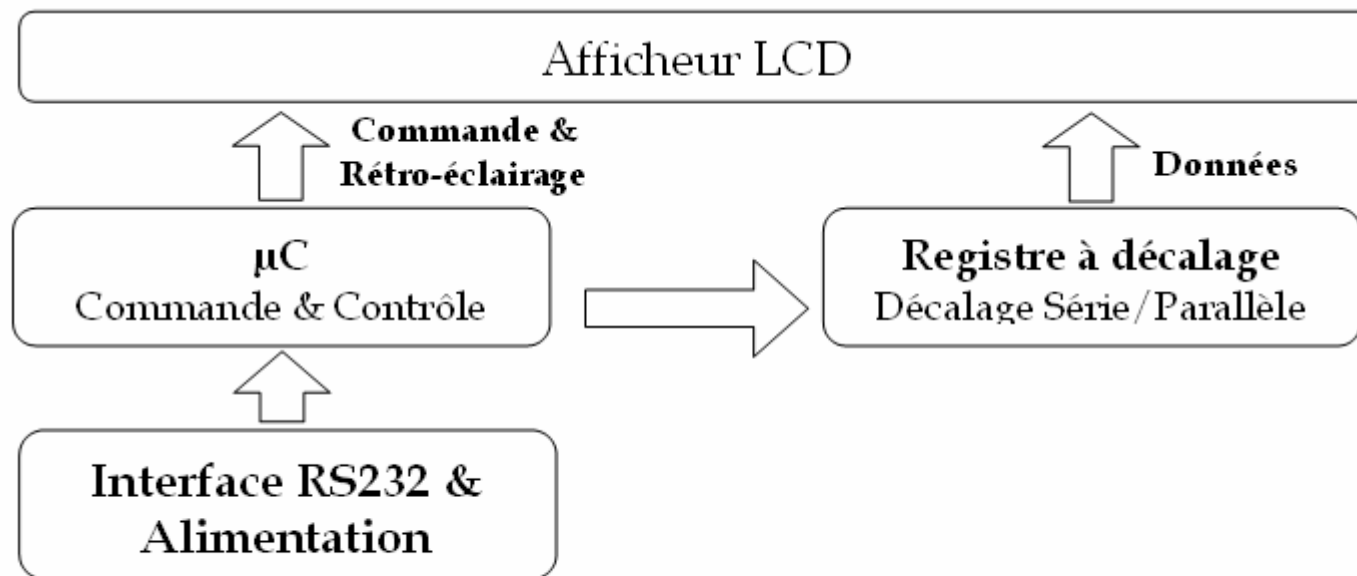


Figure 21, Vue d'ensemble de l'interface de l'afficheur LCD Série

Partie Matérielle

Commande Infrarouge (3/3)

- Vue d'ensemble

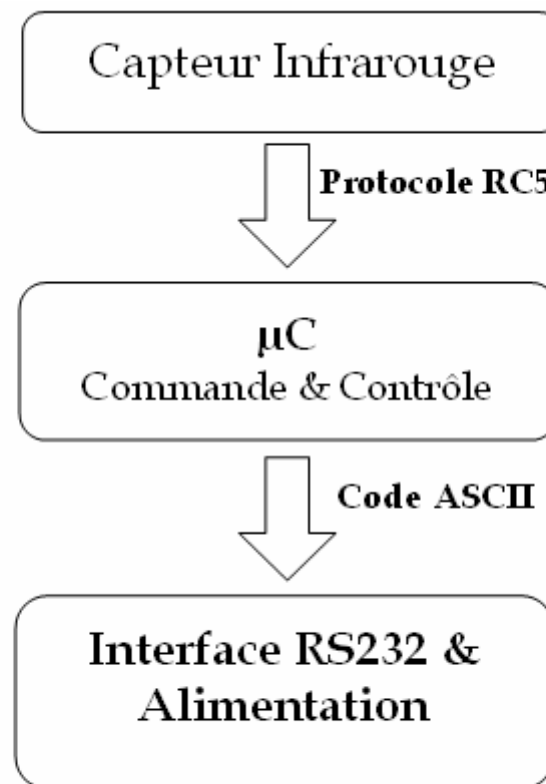


Figure 31, Vue d'ensemble du module de commande Infrarouge

Partie Matérielle

Date actuelle/Horaires de Prières (1/3)

■ Fonctions implémentées

- Déterminer la date et l'heure actuelle
- Sauvegarder les paramètres du système hors tension
- Sauvegarder les horaires de prières d'une façon permanente

■ Solutions utilisées

- Utiliser des composants basés sur la technologie I₂C
- Utiliser une horloge RTC I₂C fonctionnant sous batterie et comportant un espace mémoire (RAM) supplémentaire pour stocker les paramètres de configuration du système
- Utiliser une mémoire EEPROM pour stocker les horaires de prières

■ Contraintes

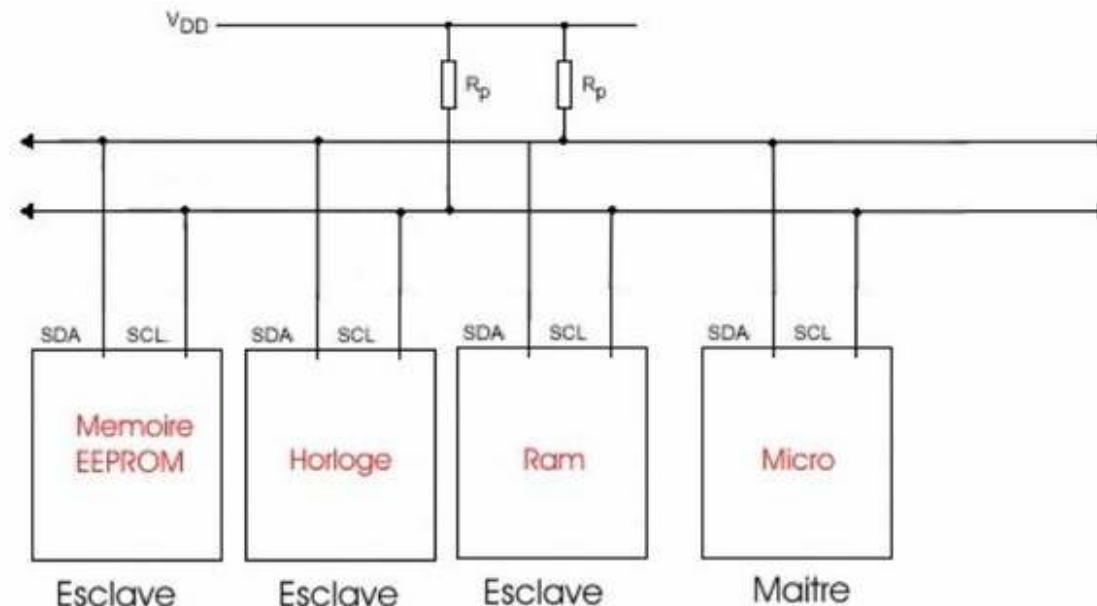
- Comprendre le fonctionnement du bus I₂C
- Connaître les adresses des composants et les commandes issues pour chacun d'entre eux

Partie Matérielle

Horaires de Prières (2/3)

■ Présentation du protocole I₂C

- Le bus I₂C (Inter-Integrated Circuit) a été inventé par Philips. C'est un bus simple basé sur uniquement deux fils. Il permet à plusieurs composants électroniques (Esclaves) connectés en cascade d'échanger des données avec un microcontrôleur (Maitre). Uniquement deux résistances pull-up sont nécessaires pour son fonctionnement.



Partie Matérielle

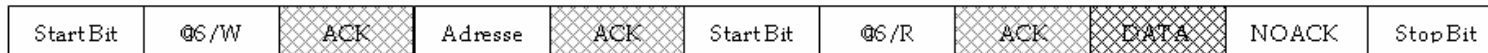
Horaires de Prières (3/3)

■ Interface I₂C

Lecture d'un octet à l'adresse courante



Lecture d'un octet à une adresse donnée



Lecture de plusieurs octets à l'adresse courante

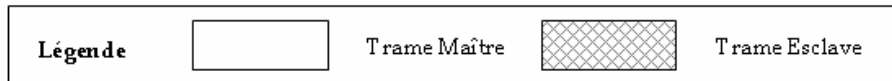


Figure 27, Protocole de lecture I₂C

Ecriture d'un octet à une adresse donnée



Ecriture de plusieurs octets à une plage d'adresse donnée

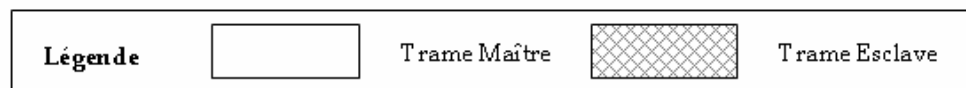


Figure 28, Protocole d'écriture I₂C



Partie Logicielle

[Plan]

- Besoins
- Présentation
 - Processus Logiciel / Outils
- Capture de besoins
- Analyse
- Conception
- Implémentation

Besoins

- Le système actuel ne dispose pas d'assez de ressources pour calculer les horaires de prières en temps réel
- Créer un utilitaire permettant de :
 - Calculer les horaires de prière
 - Stocker les données sous le format adéquat (Hex)

Présentation

Processus Logiciel/Outils (1 / 4)

- Processus Logiciel : Iconix Process
- Langage : Python
- Interface : wxPython
- SGBD : sqlite3

Présentation

Processus Logiciel/Outils (2 / 4)

■ Processus Logiciel : Iconix Process

- Iconix Process est un processus de développement logiciel basé sur le langage UML

- Il se distingue du RUP par sa simplicité.
Il suffit d'utiliser 20% de la notation UML pour faire 80% du travail.

- Il est centré sur les cas d'utilisation, **tout comme RUP.**

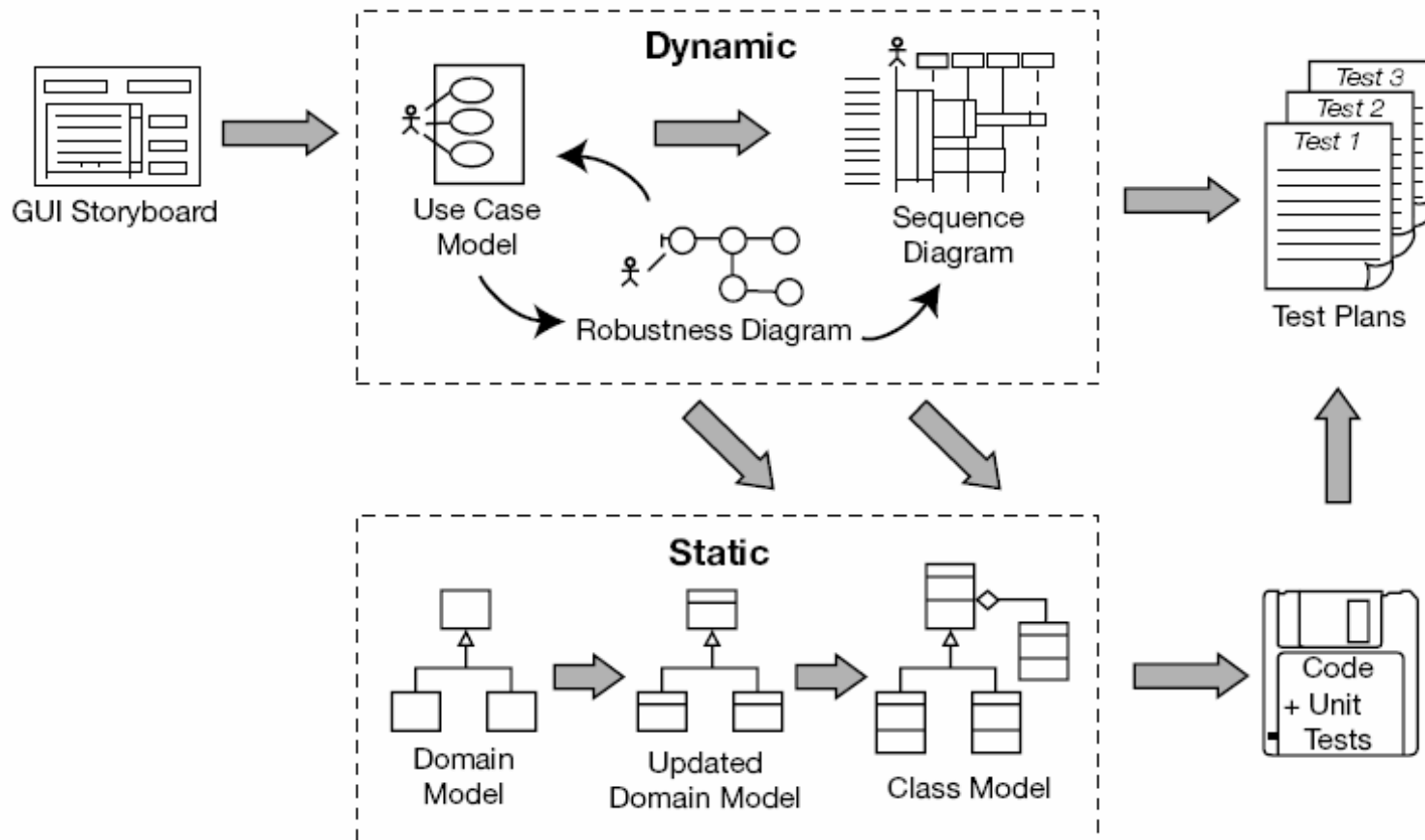
- Il comporte principalement 4 phases :

- Capture des besoins,
- Analyse,
- Conception
- Implémentation.

Présentation

Processus Logiciel/Outils (3 / 4)

■ Processus Logiciel • Iconix Process



Présentation

Processus Logiciel/Outils (4 / 4)

■ Langage Python

- Le langage Python est un langage orienté objet, interprété, multi-plateforme de l'univers des logiciels open source.

■ wxPython

- wxPython est le portage du Framework graphique (multiplateforme) wxWindow pour le langage python.

■ SQLITE 3

- Sqlite 3 est un SGBD, multi-plateforme, très léger, conçu pour les applications centrées sur un volume de données restreint.

[Conception - Capture de besoins (1 / 4)]

- Utilisateurs

- Concepteur du système

- Besoins

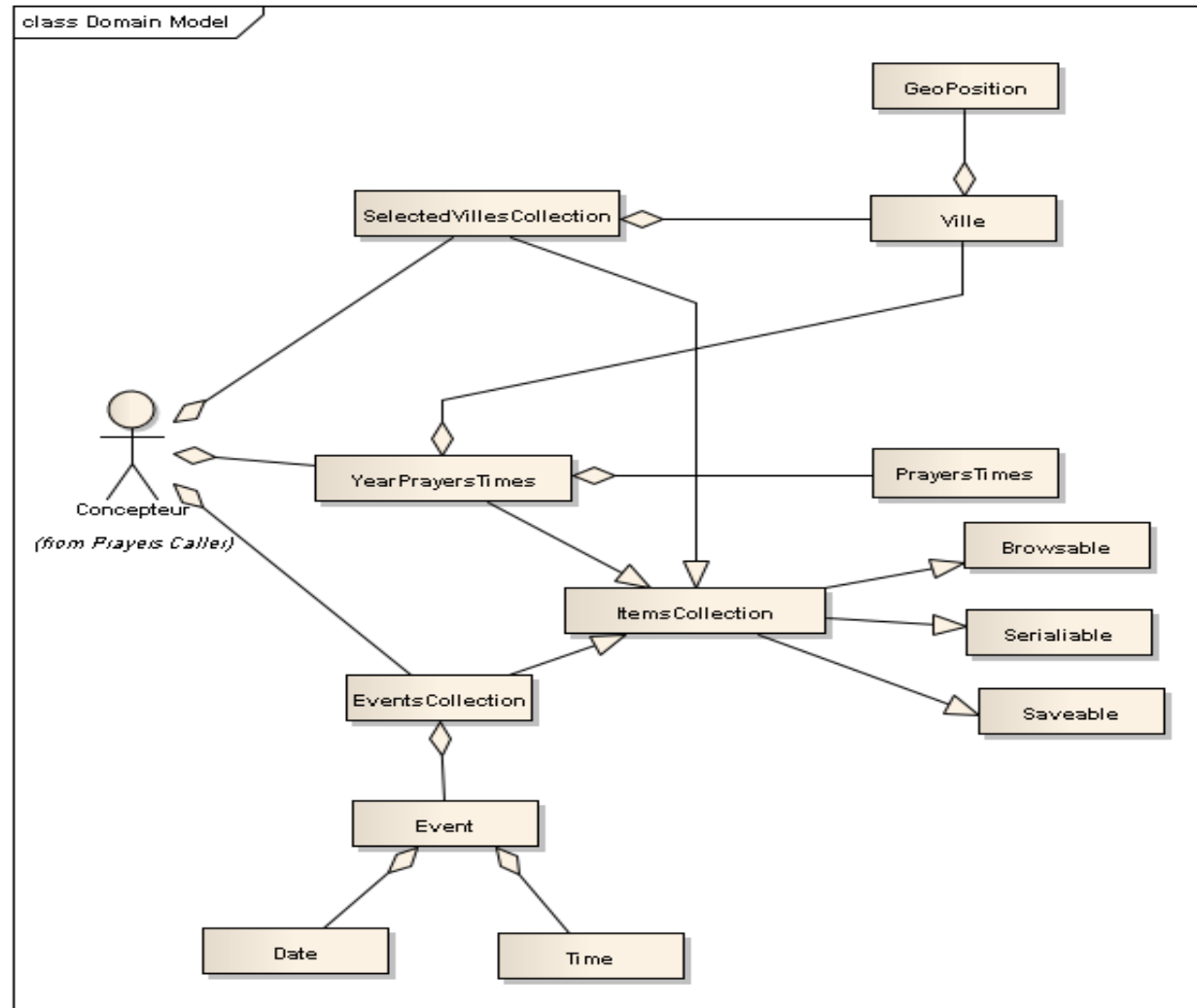
- Sélectionner 13 villes parmi les villes disponibles
- Ajuster les coordonnées et les paramètres des villes sélectionnés
- Générer et éditer les horaires de prières pour ces villes
- Générer le fichier HEX comportant les données qui devront être transférés dans la mémoire I2C de notre montage.

Conception - Capture de besoins (2/4)

- On identifie les cas d'utilisations suivants :
 - Afficher villes sélectionnées
 - Ajouter Ville
 - Ordonner villes
 - Retrouver coordonnées initiales ville
 - Sélectionner ville (liste choix)
 - Supprimer ville
 - Afficher coordonnées villes
 - Modifier coordonnées ville
 - Afficher horaires de prières
 - Générer horaires de prières
 - Modifier heures de prières
 - Générer fichier Hex
- Ces cas d'utilisations seront organisés en 5 packages

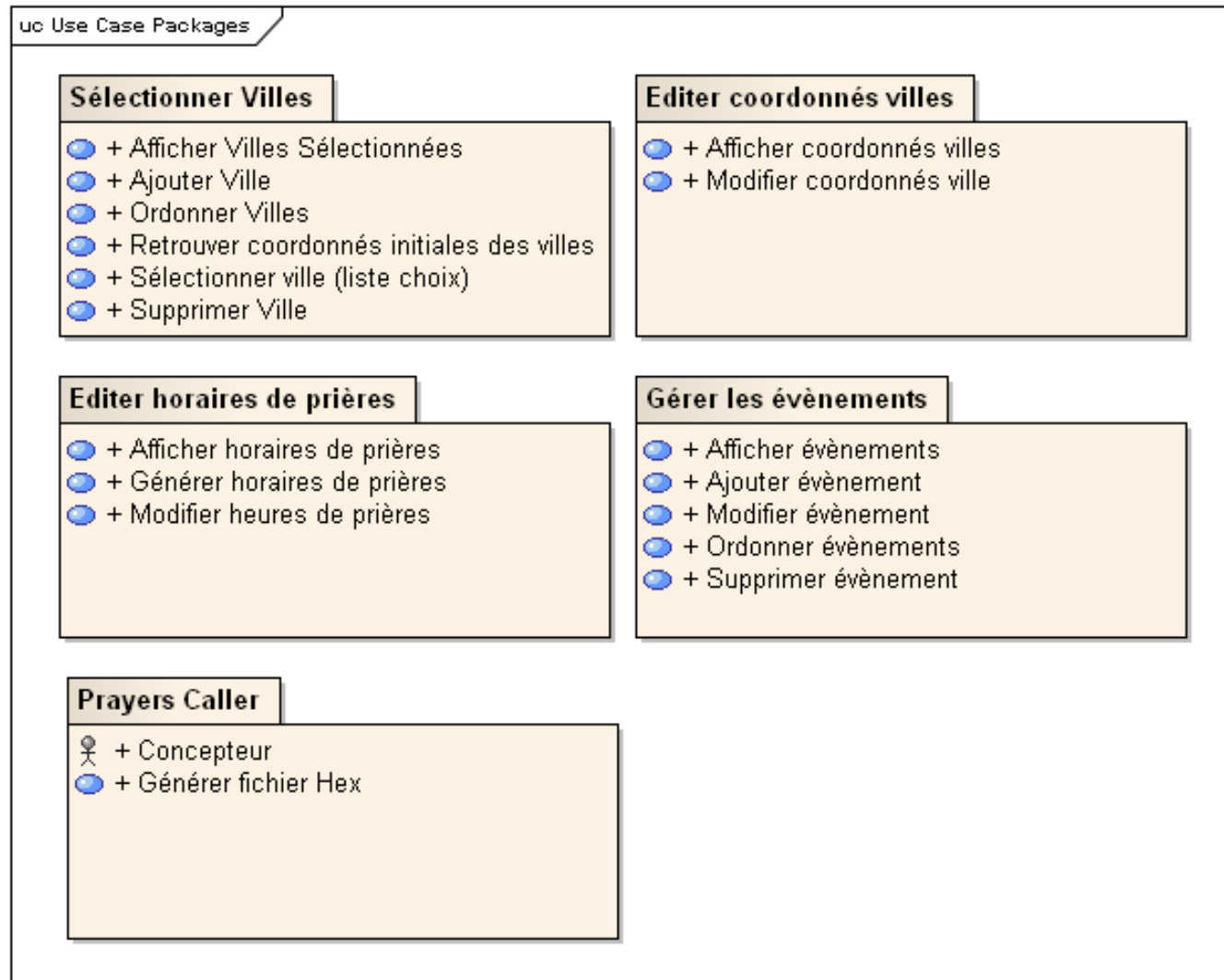
Conception - Capture de besoins (3/4)

- Modèle de domaine



Conception - Capture de besoins (4/4)

■ Packages

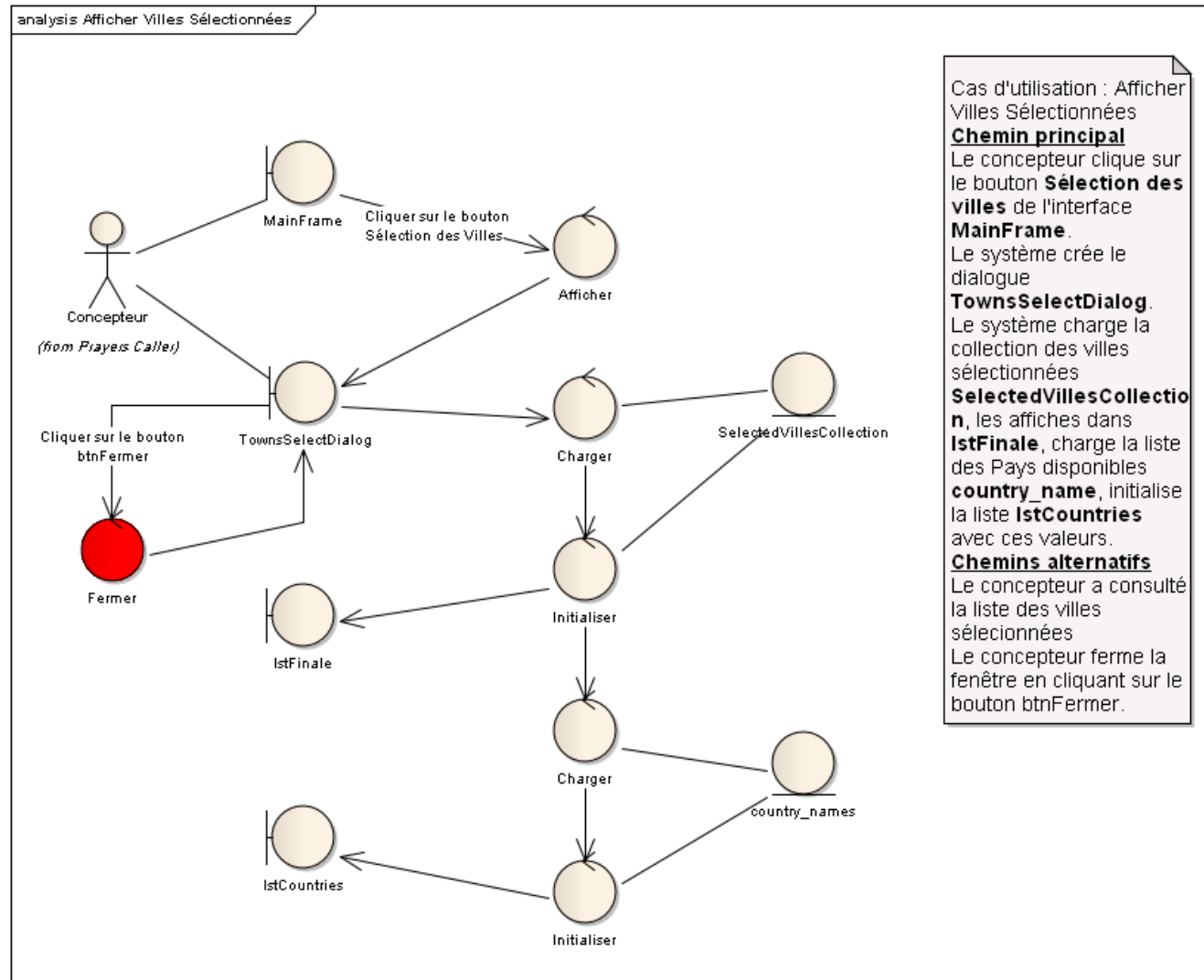


[Conception - Analyse (1 / 3)]

- Dans la phase d'analyse il s'agit de dessiner les **diagrammes de robustesses** et les **diagrammes de séquences** pour chaque cas d'utilisation.
- Les **diagrammes de robustesses** permettent de lever toute ambiguïté du texte des Use cases. L'enchaînement dans ces diagrammes est décrit par le texte figurant dans une note UML à droite dans chacune d'elles.

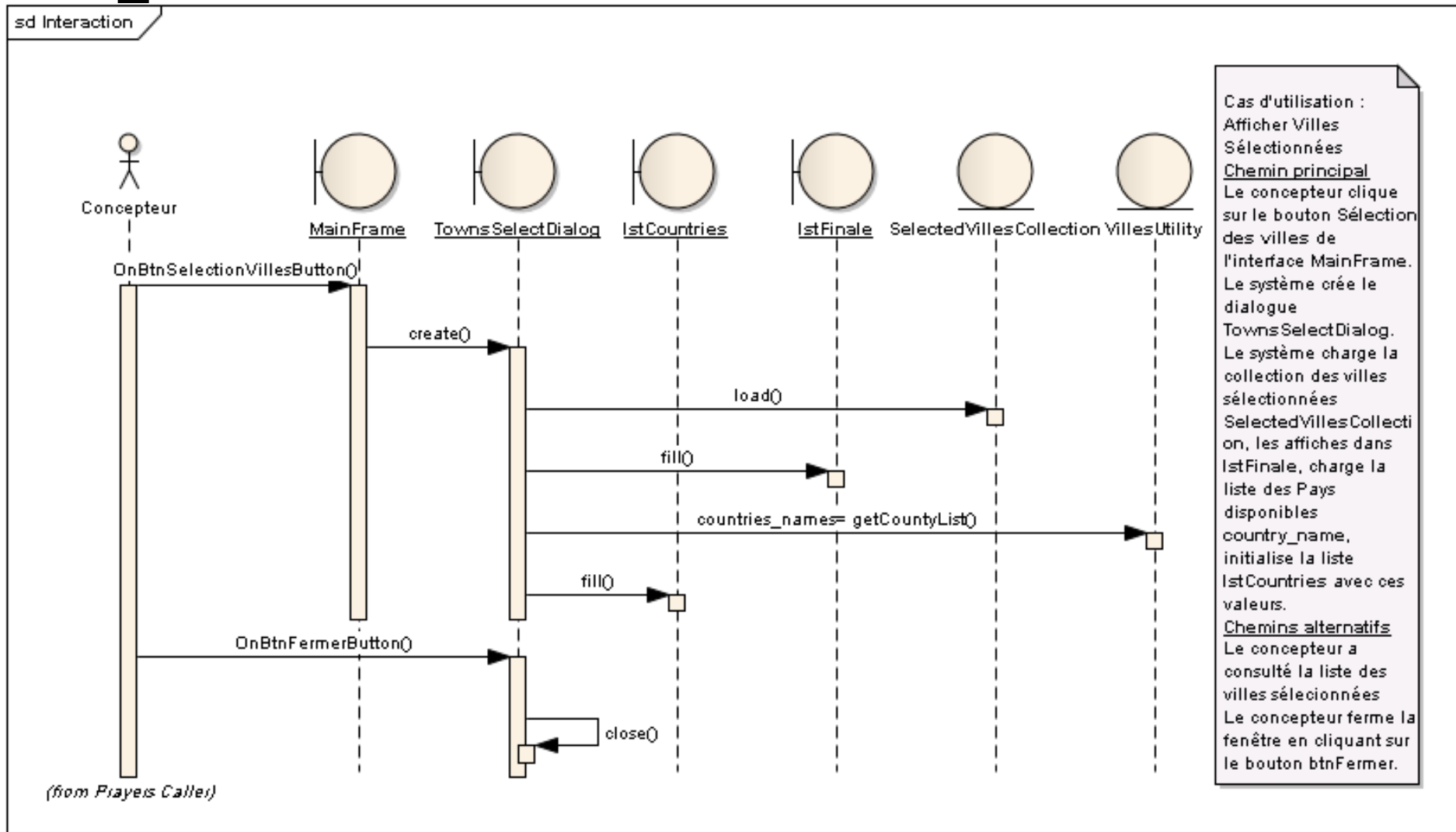
Conception - Analyse (2/3)

Diagramme de Robustesse



Conception - Analyse (3/3)

Diagramme de Séquence

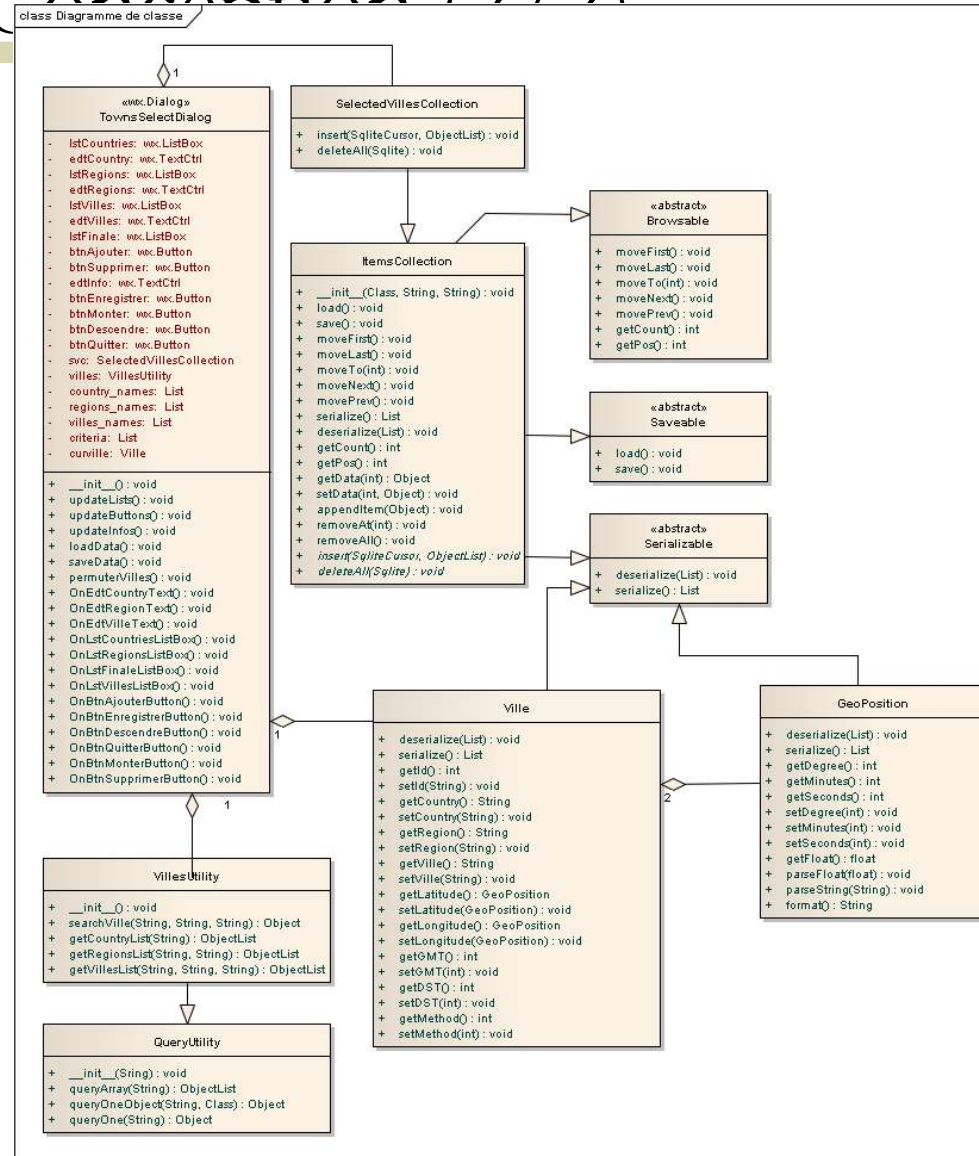


[Conception - Conception (1/2)]

- “Analysis is about building the right system. Design is about building the system right. [1]”
- Dessiner le diagramme de classes à partir du modèle de domaine, des diagrammes de séquences, des diagrammes de robustesses et des cas d'utilisation.
 - Le **nom des classes** qui serviront de squelette pour notre application provient du **modèle du domaine** ainsi que quelques attributs.
 - L'interaction entre ces classes ainsi que **les méthodes** et **les attributs** qui les constituent sont déduites des **diagrammes de robustesses**.
 - **L'ordre d'appel des méthodes** est impliqué par les **diagrammes de séquences**.

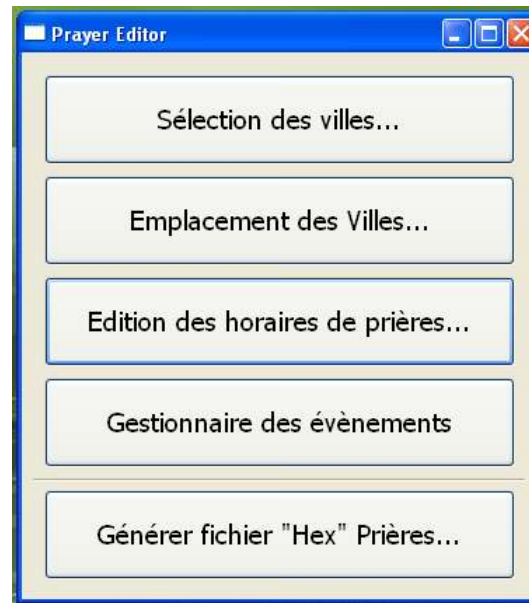
Conception - Conception (2 / 2)

Diagramme de Classes



[Conception - Implémentation (1 / 2)]

- Fenêtre Principale



Conception - Implémentation (2/2)

Données de Sélection des Villes

Sélection des Villes

Pays

Région

Ville

Afghanistan

Albania

Algeria

Andorra

Angola

Anguilla

Antarctica

Antigua_barbuda

Argentina

Armenia

Aruba

Ashmore_cartier_is

Australia

Austria

Azerbaijan

Bahamas

Bahrain

Bangladesh

Barbados

Bassas_de_india

Belarus

Belgium

Belize

Benin

Ajouter

Supprimer

Liste des villes retenues

Tunisia - Susah - Hammam Sousse

Tunisia - Susah - Sousse

Tunisia - Susah - Zaouiet Sousse

Tunisia - Safaqis - Port de Peche

Tunisia - Safaqis - Sfax

Tunisia - Safaqis - Skhira

Monter

Descendre

Enregistrer

Quitter

43

Statistiques

Coûts (1 / 3)

- Coût approximatif du prototype
 - Total [Prix des composants/Circuit imprimés] : 450 dinars
 - Module afficheur 7 segments : 30DT/Module x 8
 - Module afficheur LCD : 40 DT
 - Module lecteur MP3 : 60DT
 - Module Principal : 55DT
 - Module Alimentation : 20DT
 - Connectique : 15DT

Statistiques

Temps de Conception (2/3)

- Temps d'étude : 3 mois
- Temps de réalisation (Schéma de Principe – Simulation – Routage – Soudure - Programmation des μC) :
 - Affichage : 14 Jours
 - Lecteur MP3 : 21 Jours
 - Affichage LCD : 5 Jours
 - Télécommande Infrarouge : 21 Jours
 - Module Principal : 28 Jours
 - Utilitaire de calcul des horaires de prière : 28 Jours
- La conception de l'utilitaire de calcul des horaires de prière a été conçu en parallèle avec d'autres activités (Module Affichage, Module Principal)

Statistiques

Temps de réalisation (3/3)

- Temps approximatifs
 - Circuits imprimés en sous-traitance : 3 journées
 - Soudure des composants : 1h à 2h par module
 - Temps du montage : 2h à 3h

Conclusion (1 / 2)

- Coût très élevé → Possibilité d'amélioration et d'optimisation
 - Concevoir un seul système de commande pour les huit modules d'affichage (Gain de plus de 75%)
 - Éliminer l'afficheur LCD Série : Utiliser les afficheurs 7 segments à sa place (Gain de 10%)
 - Utiliser un lecteur Wave basé sur un μC et une mémoire SD (Gain de 20%)

Conclusion (2 / 2)

- Acquisition de nouvelles connaissances en électronique
 - **Techniques**
 - Réalisation des circuits imprimé (Schéma de principe/Routage/Simulation)
 - Soudure
 - Calculs théoriques
 - **Technologies**
 - Familles de PIC (16F et 18F)
 - I₂C
 - RS232
 - RC5
- Utilisation pratique d'un Processus de développement logiciel : **Iconix Process**
- Utilisation de logiciels Open-Source/Multi-plateforme pour une raison de portabilité :
 - Python
 - wxPython
 - SQLite

A decorative graphic consisting of a thin gold circle. A horizontal bar, colored with a gold-to-white gradient, passes through the center of the circle. The text "Merci pour votre attention" is centered on this bar. A large, thick black left square bracket is positioned to the left of the text, and a large, thick gold right square bracket is positioned to the right of the text.

Merci pour votre attention