

TRAVAUX D'ÉTUDE ET DE RECHERCHE

Application mobile d'apprentissage

Asmaa SEBIA
Lamia MÉZIANI
Dragos-Adrian SEREDINSCHI

Université d'Orléans, Faculté des Sciences
Département d'Informatique

May 2012

Au sujet du projet

- ▶ Sujet : Application mobile d'apprentissage

- ▶ Particularités :

Application mobile Pour les smartphones, tablettes

Apprentissage Trois objectifs : lire, écrire & compter

Utilisateurs Enfants de moins de 5 ans

Diversité Prise en charge pour plusieurs langages

- ▶ Nom :

Lire-Écrire-Compter (LEC)

Application mobile pour l'apprentissage

Système d'exploitation - **Android**

1. **Restrictive**, en raison du système de fichiers, ressources disponibles, mais..

Application mobile pour l'apprentissage

Système d'exploitation - **Android**

1. **Restrictive**, en raison du système de fichiers, ressources disponibles, mais..
2. **Avantageux** : écran tactile plus confortable ; plus efficace que des sites web, plus facile a comprendre et utiliser, notamment pour les enfants

Le dispositif mobile augmenter l'apprentissage de manières appropriées [Marrer G., 2009]

Application mobile pour l'apprentissage

Système d'exploitation - **Android**

1. **Restrictive**, en raison du système de fichiers, ressources disponibles, mais..
2. **Avantageux** : écran tactile plus confortable ; plus efficace que des sites web, plus facile à comprendre et utiliser, notamment pour les enfants

Le dispositif mobile augmenter l'apprentissage de manières appropriées [Marrer G., 2009]

3. **Dispositifs Android** : beaucoup (plus de 300 millions [Andy Rubin, 2012]).

Application mobile pour l'apprentissage

Système d'exploitation - **Android**

1. **Restrictive**, en raison du système de fichiers, ressources disponibles, mais..
2. **Avantageux** : écran tactile plus confortable ; plus efficace que des sites web, plus facile à comprendre et utiliser, notamment pour les enfants

Le dispositif mobile augmenter l'apprentissage de manières appropriées [Marrer G., 2009]

3. **Dispositifs Android** : beaucoup (plus de 300 millions [Andy Rubin, 2012]).
4. **Android SDK** : bien documenté et maintenu ; Open Source

Application mobile pour l'apprentissage

Particularités de l'apprentissage

Méthodes envisagées

1. Lire
2. Écrire
3. Compter

Application mobile pour l'apprentissage

Particularités de l'apprentissage

Méthodes envisagées

1. Lire
2. Écrire
3. Compter

Lire

Apprendre à lire l'alphabet, syllabes, mots, phrases d'une langue.

Application mobile pour l'apprentissage

Particularités de l'apprentissage

Méthodes envisagées

1. Lire
2. Écrire
3. Compter

Écrire

Apprendre à écrire à l'aide de la surface tactile.

Application mobile pour l'apprentissage

Particularités de l'apprentissage

Méthodes envisagées

1. Lire
2. Écrire
3. Compter

Compter

Apprendre à faire le calcul mental, distinguer les différents chiffres et opérations mathématiques de base.

Description générale du logiciel

Les cas d'utilisation

Créé en vertu de l'hypothèses que l'utilisateur (enfant) :

1. **Ne peut pas lire**

- ⇒ beaucoup d'images aussi expressives que possible
- ⇒ le logiciel doit parler pour atteindre une interactivité

Description générale du logiciel

Les cas d'utilisation

Créé en vertu de l'hypothèses que l'utilisateur (enfant) :

1. **Ne peut pas lire**

- ⇒ beaucoup d'images aussi expressives que possible
- ⇒ le logiciel doit parler pour atteindre une interactivité

2. **Comprend son langue parlée**

- ⇒ peut interpreter ce que le logiciel parle

Description générale du logiciel

Les cas d'utilisation

Créé en vertu de l'hypothèses que l'utilisateur (enfant) :

1. **Ne peut pas lire**

- ⇒ beaucoup d'images aussi expressives que possible
- ⇒ le logiciel doit parler pour atteindre une interactivité

2. **Comprend son langue parlée**

- ⇒ peut interpreter ce que le logiciel parle

3. **Assistée d'une personne** (au moins au début de l'utilisation) -
car il pourrait être difficile de naviguer

Description générale du logiciel

Les cas d'utilisation

Créé en vertu de l'hypothèses que l'utilisateur (enfant) :

1. **Ne peut pas lire**

⇒ beaucoup d'images aussi expressives que possible

⇒ le logiciel doit parler pour atteindre une interactivité

2. **Comprend son langue parlée**

⇒ peut interpreter ce que le logiciel parle

3. **Assistée d'une personne** (au moins au début de l'utilisation) -
car il pourrait être difficile de naviguer

4. **A le désir d'apprendre et d'expérimenter** 😊

Description générale du logiciel

Les cas d'utilisation

Ainsi, en dehors de la spécification du projet, nous avons également diverses :

Description générale du logiciel

Les cas d'utilisation

Ainsi, en dehors de la spécification du projet, nous avons également diverses :

Contraintes

- ▶ Navigation facile : « Keep It Simple »
- ▶ Faire parler tout l'information présentée

Description générale du logiciel

Les cas d'utilisation

Ainsi, en dehors de la spécification du projet, nous avons également diverses :

Contraintes

- ▶ Navigation facile : « Keep It Simple »
- ▶ Faire parler tout l'information présentée

Exigences

- ▶ Chaque langage est différent, donc prêt de parler en plusieurs langages.
- ▶ Différents niveaux de difficulté pour tous les activités
- ▶ Persister les progrès réalisés sur une activité

Description générale du logiciel

Entites du logiciel

- ▶ On peut observé qu'une activité de apprentissage est caractérisé par plusieurs attributs :
 1. Le **langage** de parler (français, anglais, etc..)
 2. La **méthode** d'apprentissage (lire, écrire, compter)
 3. Le **niveau** de difficulté (plus bas, plus haut, etc..)
 4. L'activité elle-même - l'**exercice** qui l'utilisateur est en train a faire
 5. Solutions possibles pour l'exercice courant - **choix**
- ⇒ les entites du logiciel (modèle)

Description générale du logiciel

Entites du logiciel - relations

- Une approche hiérarchique :

Langage $\xrightarrow{3}$ Méthode $\xrightarrow{1..*}$ Niveau $\xrightarrow{1..*}$ Exercise $\xrightarrow{4}$ Choix

- Remarques :

- 3 Méthodes pour chaque langage, parce que nous avons trois objectifs : Lire, Écrire, Compter
- 4 Choix pour chaque Exercise : l'utilisateur doit choisir une de quatre réponses possibles, seulement une est correcte

Architecture du logiciel

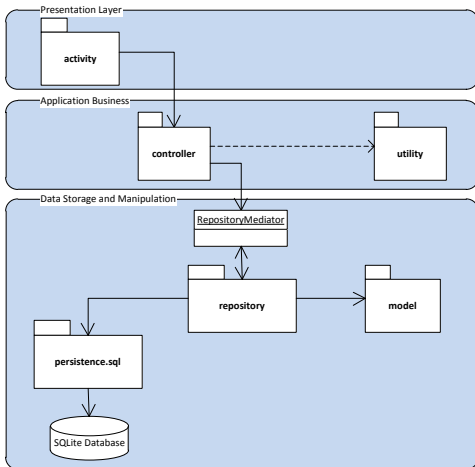


Figure: Découpage modulaire

Architecture du logiciel

MVC

Méthode de conception : **Model-View-Controller**

- ▶ Quelques avantages :
 1. Un plus d'indépendance entre les composants
 2. Meilleur choix pour notre logiciel qui est centré sur l'utilisateur
 3. Facilite le développement en l'équipe
- ▶ Quelques remarques :
 - ▶ View : instances de Android Activity classe
 - ▶ Model : lié entre eux par une objet intermédiaire - RepositoryMediator - s'assure que les relations sont cohérentes et persistantes

Architecture du logiciel

MVC

Selon le patron MVC, on peut identifier trois couches pour notre application :

Architecture du logiciel

MVC

Selon le patron MVC, on peut identifier trois couches pour notre application :

View

- ▶ Les activités et les interfaces avec l'utilisateur
- ▶ Activity classes, et les XML fichiers correspondantes qui décrit la structure de l'interface
- ▶ Techniques de parler
- ▶ Chaque View est lié avec un objet Controlleur - pour obtenir les données
- ▶ une Activity spéciale : MainActivity - la classe principale qui démarre l'application

Architecture du logiciel

MVC

Selon le patron MVC, on peut identifier trois couches pour notre application :

Controlleur

- ▶ Toutes les affaires du logiciel
- ▶ Fait le lien entre les données et la présentation
- ▶ Personnalisation basée sur la langage sélectioné
- ▶ L'accès aux données se fait par le RepositoryMediator
- ▶ Classe de base abstraite : BasicLECController

Architecture du logiciel

MVC

Selon le patron MVC, on peut identifier trois couches pour notre application :

Model (couche de données)

- ▶ La plus bas niveau - tout ce qui touche aux données
- ▶ Comprend plusieurs paquets : Model, Repository, Persistence
- ▶ Fait le lien avec la base de données : SQLite
- ▶ La base de données est initialisé lorsque le premier démarrage de l'application, à partir d'un fichier XML qui décrit le contenu et les relations de nos modèles

Points techniques

Lire

- ▶ Le **synthèse vocale** est réalisé avec le paquet `android.speech.tts.TextToSpeech` qui peut être utilisé pour plusieurs langages - nous avons choisi d'implémenter le logiciel avec 3 langages predefinis : *Anglais, Français, Allemand*
- ▶ Une fois que l'utilisateur a choisit son langage, toute l'interaction est faite dans le langage respective
- ▶ Les exercice de lire sont les plus faciles - l'utilisateur doit juste reconnaître le lettre / syllabe / mot présenté par l'application
- ▶ Une échelle présente les progrès accomplis jusqu'à présent (à l'intérieur du même niveau)

Points techniques

Compter

- ▶ Les exercices de compter ont la même format que les exercices de lecture, mais l'utilisateur est présenté avec des diverses opérations mathématiques simples
- ▶ Nous avons identifié des problèmes avec `android.speech.tts.TextToSpeech`, pour les opérations de différence (e.g. $4 - 1 = 3$), l'operator « $-$ » n'est pas prononcé correctement

Points techniques

Écrire

Écriture

- ▶ Normalement, les activités d'écriture se composer de plusieurs exercices dans lequel l'enfant essayer de faire les symboles données comme exemple
⇒ dans une tablette, l'enfant doit essayer de la même manière et le logiciel doit **reconnaître** si l'écriture est correct ou pas

Pas facile !

- ▶ Les autres points techniques sont semblable avec les ceux des l'autres types d'activités

Exemples de fonctionnement

Page d'accueil

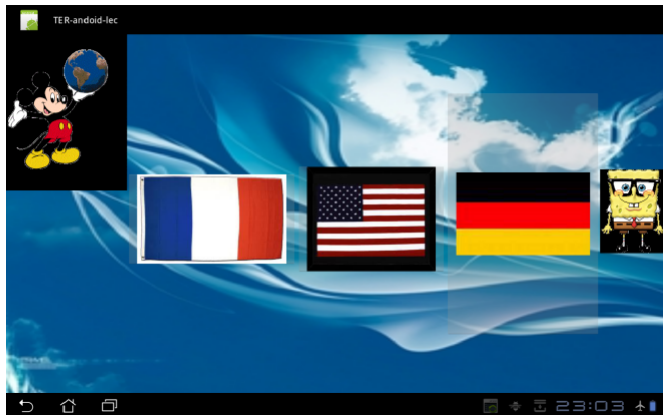


Figure: La première page contient 3 drapeaux et chaque drapeau représente une langue d'apprentissage.

Exemples de fonctionnement

Presentation des méthodes

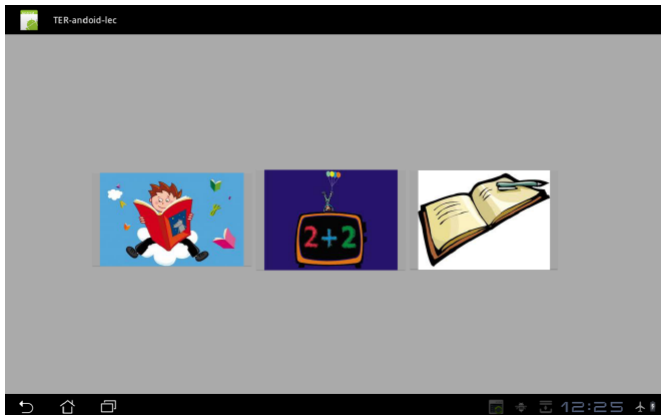


Figure: La deuxième page contient les 3 méthodes disponibles.

Exemples de fonctionnement

Les niveaux

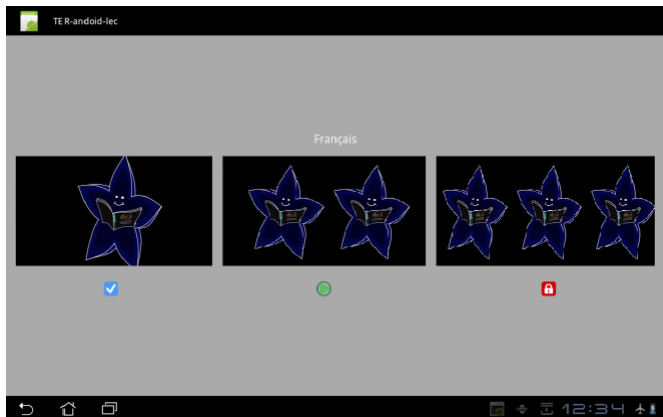


Figure: Après une méthode est choisit, les niveaux sont présentées. Ici, l'utilisateur a complété le première niveau.

Exemples de fonctionnement

Une exercise

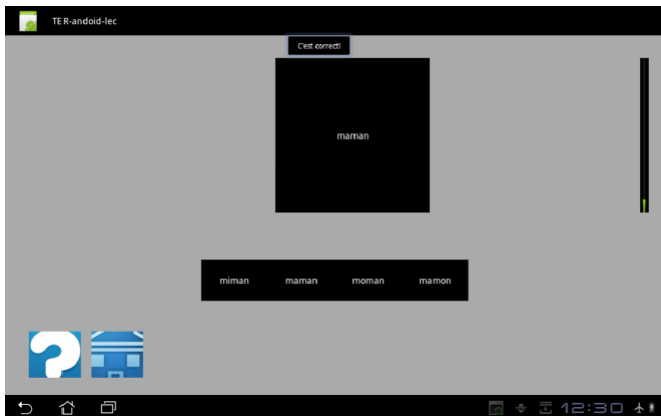


Figure: Exemple de fonctionnement d'une exercise de type *Lire*, pour le langage *Français*.

Extensions possibles

1. Le logiciel est facile à étendre - pour ajouter des autres langages :
 - 1.1 Modifier juste le fichier XML qui décrit la base de données
 - 1.2 faire de la place (avec le drapeau) dans la page d'accueil
2. Le méthode d'écrire !
3. Multi-utilisateurs : Création des comptes pour les enfants et les instituteurs
 - 3.1 Modifier le fichier XML qui décrit la base de données
 - 3.2 Ajouter les classes de modèles, view & contrôleur nécessaires
4. Personnaliser l'échelle : plus convivial - suffit de modifier le widget qui décrit l'échelle

Bibliographie



Marrer G.

A Mobile Learning Definition for My Campus, 2009.



Andy Rubin

Android@Mobile World Congress : It's all about the ecosystem,
2012.