

Correction du TP Java ME MIDP

A été distribué :

- a) `sun_java_wireless_toolkit-2_5_2-windows.exe`, (l'environnement de développement de SUN pour portable le "Wireless Toolkit 2.5.2 for CLDC" que l'on trouve à <http://java.sun.com/products/sjwtoolkit/download.html> pour Linux et Windows) pour windows. Cet environnement est appelé le WTK.
- b) L'énoncé de cet exercice (ce que vous lisez en ce moment !)
- c) Le cours dans `JavaMEComplet.pdf`

- 1°) Installer l'environnement WTK (par exemple, sous `C:\Applications\WTK`)
- 2°) Lancer cet environnement sur les démonstrations proposées par exemple : `Demos`, `CityGuide`, `UIDemo` par **OpenProject** puis **Run**. Votre souris ne doit servir qu'à appuyer sur les touches de l'émulateur.
- 3°) Changer de **Device** (de **DefaultColorPhone** à **QuertyDevice**) et tester sur certaines midlets.

L'environnement "Wireless Toolkit"

Exercice 1 :

- 4°) Ecrire une MIDlet qui affiche bon courage.
- 4.1) Pour cela cliquer **New Project**, Entrer un nom de projet (`EDJ2MECNAM`) et un nom de classe (`IHMcnam`) puis cliquer **Create Project**. Accepter les configuration par défaut dans la fenêtre suivante : cliquer **OK**.
- 4.2) Vérifier qu'un nouveau projet a été créer : le répertoire `EDJ2MECNAM` a été créé sous `C:\Documents and Settings\nomDUtilisateurDeVotreMachine\j2mewtk\2.5.2\apps`, (pour Windows XP).
- 4.3) Dans ce répertoire `EDJ2MECNAM` ont été créés plusieurs fichiers et sous-répertoires. Editer votre MIDlet dans le fichier `IHMcnam.java` du répertoire `src`. Ecrire cette MIDlet

Voici le code d'une telle MIDlet :

```
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.TextBox;
import javax.microedition.lcdui.TextField;
import javax.microedition.midlet.MIDlet;

public class IHMcnam extends MIDlet {
    private static final int MAX_TEXT_SIZE = 64;
    protected TextBox textBox;
    protected Display display;
    protected boolean started;

    protected void startApp() {
        if (!started) {
            String str = null;
            // str = "Second programme avec TextBox";
            str = "Bon courage";
            textBox = new TextBox("TextBox Example", str,
                MAX_TEXT_SIZE, TextField.ANY);
            display = Display.getDisplay(this);
            display.setCurrent(textBox);
            started = true;
        }
    }
}
```

```

protected void pauseApp() {
}

protected void destroyApp(boolean unconditional) {
}
}

```

4.4) Construire la MIDlet par l'environnement WTK en cliquant **Build**. Lancer l'exécution en cliquant **Run**.

IHM haut niveau

Exercice 2 :

5°) Ecrire une MIDlet d'authentification qui affiche un login et un mot de passe et une Command. Lorsque l'utilisateur a rempli ces 2 champs, il doit cliquer la Command. Si le couple (login, mot de passe) est correct (par exemple mot de passe = login suffixé par pw), on passe à un écran suivant (celui de l'exercice précédent par exemple).

Cette MIDlet d'authentification doit présenter l'interface graphique suivante :



Pour cela, créer un seconde MIDlet dans la suite de MIDlet EDJ2MECNAM. Utiliser le WTK. Cliquez **Settings...** Dans la fenêtre suivante, cliquer **Add**. Remplir les champs :

- **Name** par **PhaseAuth**
- **Icon** par **PhaseAuth.png**
- **Class** par **PhaseAuth**

Cliquez **OK**. Une nouvelle MIDlet doit apparaître dans la suite de MIDlets

6°) Ecrire la partie authentification de cette seconde MIDlet dans le fichier `PhaseAuth.java`. Utiliser la documentation CLDC/MIDP que vous avez en local. Au fait où est elle ?

Voici le code complet d'une telle MIDlet pour les deux questions 5°) et 6°) :

```

import javax.microedition.lcdui.*;
import javax.microedition.midlet.MIDlet;

public class PhaseAuth extends MIDlet implements CommandListener {
    private static final int MAX_TEXT_SIZE = 12;
    protected TextBox textBox;
    protected Command backCommand;

    protected Form laForme;
    protected TextField textFieldLogin;
    protected TextField textFieldPassword;
    protected Command next;

    protected Alert uneAlerte;
    protected Display display;
    protected boolean started;

    protected void startApp() {
        if (!started) {
            creeEcrans();

            display = Display.getDisplay(this);
            display.setCurrent(laForme);
            started = true;
        }
    }

    public void creeEcrans() {
        // La Form
        textFieldLogin = new TextField("Login : ", "",
            MAX_TEXT_SIZE, TextField.ANY);
        textFieldPassword = new TextField("Mot de passe : ", "",
            MAX_TEXT_SIZE, TextField.PASSWORD);
        laForme = new Form("Fenêtre d'authentification");
        next = new Command("Next", Command.OK, 0);
        laForme.append(textFieldLogin);
        laForme.append(textFieldPassword);
        laForme.addCommand(next);
        laForme.setCommandListener(this);

        // La TextBox
        String str = null;
        str = "Bravo";
        textBox = new TextBox("TextBox Example", str,
            MAX_TEXT_SIZE, TextField.ANY);
        backCommand = new Command("Back", Command.OK, 0);
        textBox.addCommand(backCommand);
        textBox.setCommandListener(this);

        // L'alerte
        uneAlerte = new Alert("",
            "Ce n'est pas le bon mot de passe\nRecommencez",
            null, AlertType.INFO);
        //uneAlerte.setTimeout(5000);
        uneAlerte.setTimeout(Alert.FOREVER);
    }

    public void commandAction(Command cmd, Displayable d) {
        if (cmd == next) {
            // System.out.println("dans commandAction");
        }
    }
}

```

```

        if
        (textFieldPassword.getString().equals(textFieldLogin.getString() + "pw")) {
            display.setCurrent(textBox);
        } else {
            display.setCurrent(uneAlerte);
        }
    } else if (cmd == backCommand) {
        // System.out.println("dans if (cmd == backCommand)");

        display.setCurrent(laForme);
    }
}

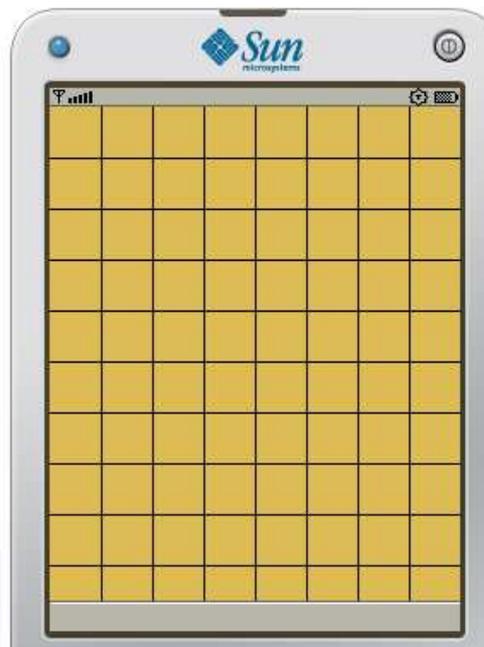
protected void pauseApp() {
}

protected void destroyApp(boolean unconditional) {
}
}

```

IHM bas niveau

7°) Ecrire une MIDlet qui présente un GoBan c'est à dire un plateau de jeu pour jouer au Go.
La couleur de fond du GoBan sera 0xDCB35C. On doit obtenir :
On devra avoir un écran comme :



La code de la MIDlet est :

```

import javax.microedition.lcdui.Display;
import javax.microedition.midlet.MIDlet;
import javax.microedition.midlet.MIDletStateChangeException;

public class GoBanMIDlet extends MIDlet {
    private boolean started;
    protected Display display;;
}

```

```

    protected void destroyApp(boolean arg0) throws
MIDletStateChangeException { }

    protected void pauseApp() { }

    protected void startApp() throws MIDletStateChangeException {
        if (!started) {
            GoBan gb = new GoBan();

            display = Display.getDisplay(this);
            display.setCurrent(gb);
            started = true;
        }
    }
}

```

Le code du Canvas GoBan est :

```

import java.util.Enumeration;
import java.util.Vector;

import javax.microedition.lcdui.Canvas;
import javax.microedition.lcdui.Graphics;

public class GoBan extends Canvas {
    private int foreground = 0xDCB35C;
    private int black = 0x000000;
    private int white = 0xFFFFFFFF;
    private int widthGoBan;
    private int heigthGoBan;

    private int largeurCarre;

    public GoBan() {
        widthGoBan = getWidth();
        heigthGoBan = getHeight();
    }

    protected void paint (Graphics g) {
        g.setColor(foreground);
        g.fillRect(0, 0, widthGoBan, heigthGoBan);
        dessineTraits(g);
    }

    void dessineTraits(Graphics g){
        int minHauteurLargeur = widthGoBan > heigthGoBan ? heigthGoBan :
widthGoBan;
        largeurCarre = minHauteurLargeur / Constantes.NOMBRE_MIN_CARRE;
        g.setColor(black);
        // dessin des lignes horizontales
        for (int i = largeurCarre ; i < heigthGoBan; i += largeurCarre){
            g.drawLine(0, i, widthGoBan, i);
        }
        // dessin des lignes verticales
        for (int j = largeurCarre ; j < widthGoBan; j += largeurCarre){
            g.drawLine(j, 0, j, heigthGoBan);
        }
    }
}

```

On a utilisé une interface Java contenant une constante :

```
public interface Constantes {  
    public int NOMBRE_MIN_CARRE = 8;  
}
```

8°) Amener à ce GoBan après authentification c'est à dire après l'exécution correcte de la MIDlet de la question 6°)

Il suffit, lorsque l'authentification est assurée, de remplacer

```
display.setCurrent(textBox);
```

par

```
display.setCurrent(gb);
```

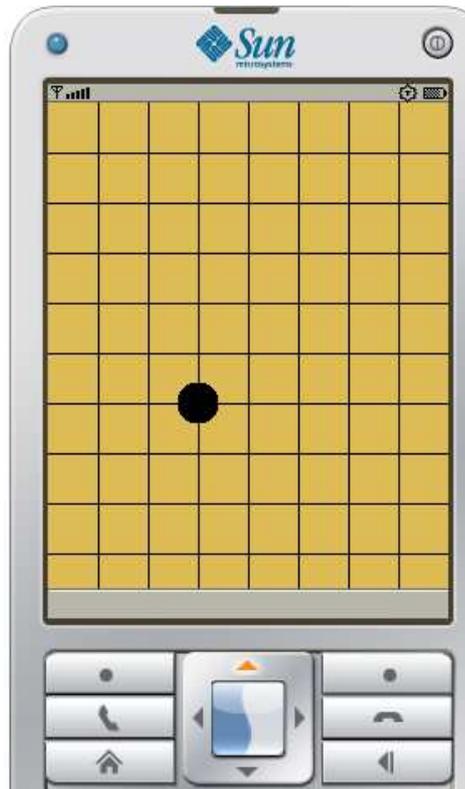
dans le code de la MIDlet d'authentification

Bonus

Tout a été codé mais la correction n'a pas été donnée (ni rédigée).
Les étudiants n'ont pas eu le temps de faire cette partie de l'exercice

9°) Faire afficher une image de pierre du jeu de go (blanche ou noire) en bas à droite du GoBan, lorsque l'utilisateur appuie sur le bouton "fire"

10°) Ecrire le code qui fait déplacer la pierre courante sur les intersections des lignes avec les touches UP, DOWN, LEFT, RIGHT. On doit avoir :



11°) Ecrire le code qui fait réafficher toutes les autres pierres déjà posées.

12°) Alternier la création des pierres (blanches puis noires) et ... jouer au go.