

# Mini Projet Java - Horasis

...

Nicolas GAILLARD  
Quentin LEVAVASSEUR  
Valentin BOUCHEVREAU  
Anthony GRIFFON



UNIVERSITÉ DE NANTES



# Ordre du jour

- Présentation des modules
- Le File Manager
- La conversion des PDF
- Le SDK Cardboard
- OpenGL et les textures
- Méthodes de travail
- Conclusion
- Démonstration



# Horasis Project

see otherwise

# Présentation des modules



*Modèle*



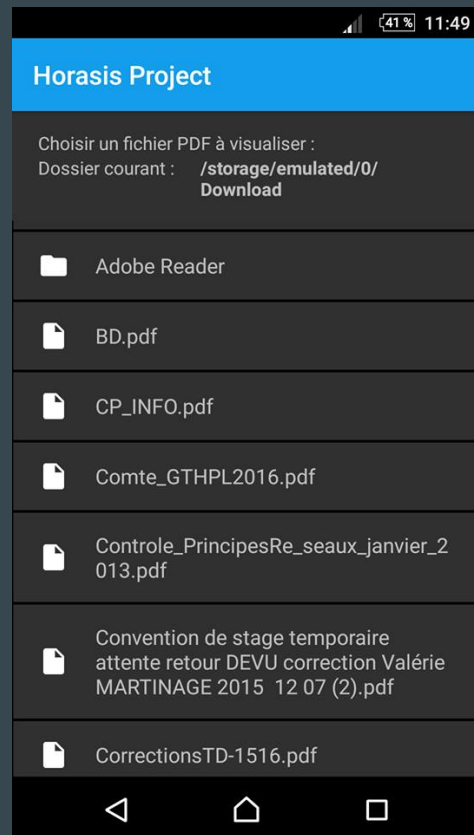
*Vue*



*Controlleur*

# Le File Manager

- MVC
  - modèle : `FolderOrPDF` (= un item de la liste)
  - vue : `RecyclerView` (= la liste)
  - contrôleur : `RecyclerViewFolderPDF`  
(= adapter : gère les fichiers)
- 2 layouts :
  - disposition de la liste des dossiers et fichiers
  - disposition d'un item de la liste (un dossier ou un fichier)



# Le File Manager

## Comment ça marche ?

- Dans l'activité :
  - un thread récupère la liste des éléments du dossier "Downloads"
  - une fois finit, il indique à l'adapter qu'il peut les traiter
  - l'adapter ajoute les éléments dans la liste (via des ViewHolder)
  - l'adapter rafraîchit ensuite la liste qui s'affiche sur le smartphone

## ListFilesActivity

```
41 values = new ArrayList<FolderOrPDF>();
42 new Thread(new Runnable() {
43     @Override
44     public void run() {
45         // Use the current directory as title
46         path = Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY_DOWNLOADS).getAbsolutePath();
47         if (getIntent().hasExtra("path")) {
48             path = getIntent().getStringExtra("path");
49         }
50         textViewPath = (TextView) findViewById(R.id.text_view_path);
51         textViewPath.setText(path);
52
53         // Read all files sorted into the values-array
54         File dir = new File(path);
55         if (!dir.canRead()) {
56             setTitle(getTitle() + " (inaccessible)");
57         }
58         String[] list = dir.list();
59         if (list != null) {
60             for (String file : list) {
61                 if (!file.startsWith(".")) {
62                     FolderOrPDF filename;
63                     if (path.endsWith(File.separator)) {
64                         filename = new FolderOrPDF(path + file);
65                     } else {
66                         filename = new FolderOrPDF(path + File.separator + file);
67                     }
68                     if (filename.getPath().endsWith(".pdf")) {
69                         filename.setType("pdf");
70                         values.add(filename);
71                     }
72                     else if (new File(filename.getPath()).isDirectory()) {
73                         filename.setType("folder");
74                         values.add(filename);
75                     }
76                 }
77             }
78         }
79         Collections.sort(values);
80         mAdapter.setFolderOrPDF(values);
81         //Deposer le Runnable dans la file d'attente de l'UI thread
82         if (this != null) {
83             runOnUiThread(new Runnable() {
84                 @Override
85                 public void run() {
86                     //code execute par l'UI thread
87                     mAdapter.notifyDataSetChanged();
88                 }
89             });
90         }
91     }
92 }).start();
```

# La conversion des PDF



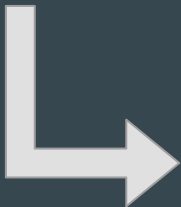
Document non exploitable sans traitement au préalable

- Création d'une classe qui effectuera les traitements
- `public final class PdfRenderer extends Object implements AutoCloseable`
- Objet qui contient l'ensemble du document
- disponible nativement dans l'API d'Android `android.graphics.pdf` depuis l'API 19
- `class PdfRenderer.page` : accès à une page du document

# La conversion des PDF

## Comment ça marche ?

1. Création d'un bitmap qui accueillera le document
2. Accès à la page du document souhaitée
3. On “colle” la page du document sur le bitmap à l'aide de la fonction `render()`



```
// create a new renderer
PdfRenderer renderer = new PdfRenderer(getSeekableFileDescriptor());

// let us just render all pages
final int pageCount = renderer.getPageCount();
for (int i = 0; i < pageCount; i++) {
    Page page = renderer.openPage(i);

    // say we render for showing on the screen
    page.render(mBitmap, null, null, Page.RENDER_MODE_FOR_DISPLAY);

    // do stuff with the bitmap

    // close the page
    page.close();
}

// close the renderer
renderer.close();
```

**Conversion**

# Le SDK Cardboard

- CardboardActivity
  - affiche deux fois l'environnement
  - fait appel aux bibliothèques Cardboard
  - dessine son environnement avec OpenGL
- Overlay (optionnel)
  - affiche deux fois des objets 2D dans l'oeil de l'utilisateur
  - pratique pour un curseur (“reticle”) ou des messages/informations





# Le SDK Cardboard

- `isLookingAtObject()`
  - calcule une marge d'erreur autour du point de référence de l'objet



- `onCardboardTrigger()`
  - manipulation de l'environnement après appui

**Objet3D**

```
@Override
public boolean isLookingAtObject(float[] headView) {
    float[] initVec = {0, 0, 0, 1.0f};
    float[] objPositionVec = new float[4];

    // Convert object space to camera space. Use the headView from onNewFrame.
    Matrix.multiplyMM(modelView, 0, headView, 0, model, 0);
    Matrix.multiplyMV(objPositionVec, 0, modelView, 0, initVec, 0);

    float pitch = (float) Math.atan2(objPositionVec[1], -objPositionVec[2]);
    float yaw = (float) Math.atan2(objPositionVec[0], -objPositionVec[2]);

    Log.i("FEUILLE", String.valueOf(Math.abs(pitch) < PITCH_LIMIT && Math.abs(yaw) < YAW_LIMIT));

    return Math.abs(pitch) < PITCH_LIMIT && Math.abs(yaw) < YAW_LIMIT;
}
```

**CardboardActivity**

```
@Override
public void onCardboardTrigger() {
    Log.i(TAG, "onCardboardTrigger");

    if (this.feuille.isLookingAtObject(headView)) {
        this.overlayView.show3DToast("Looking at Feuille");
        if (this.feuille.isOnTable()) {
            this.feuille.isOnTable=false;
        }
    }
}
```

# OpenGL et les textures

- OpenGL : Librairie graphique
  - permet d'afficher des images dans un environnement 3D

- Processus de création :

```
@Override  
public void make() {
```

⇒ initialisation des coordonnées

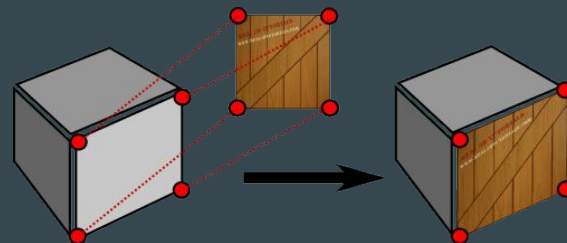
```
@Override  
public void program(int vertexShader, int passthroughShader) {
```

⇒ application du “fragment shader”

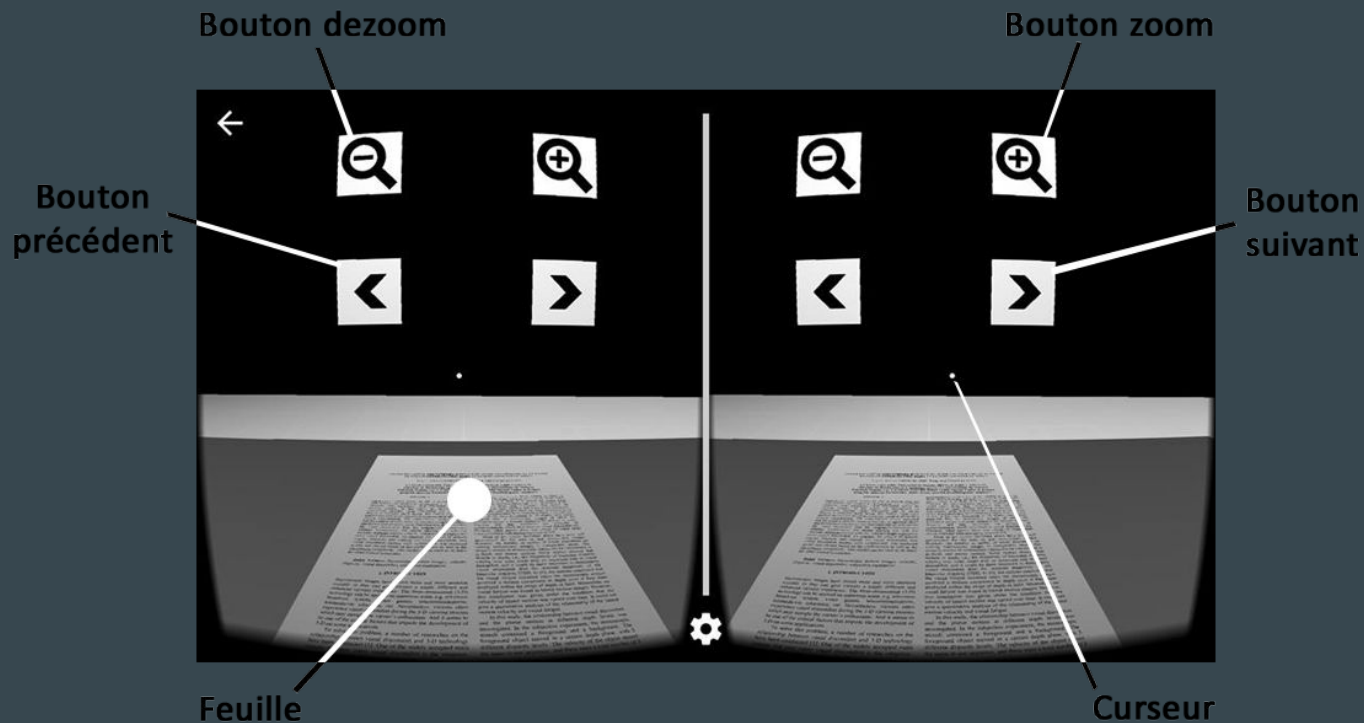
```
public static int loadTexture(Bitmap bitmap) {
```

⇒ passage de l'image en paramètre

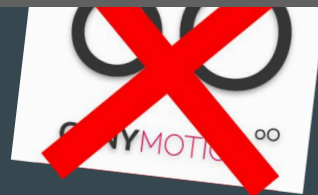
```
@Override  
public void draw(float[] lightPosInEyeSpace, float[] modelView, float[] modelViewProjection, float[] headView) {
```



# OpenGL et les textures



# Les méthodes de travail



# Conclusion

- Découverte du monde d'Android, Google Cardboard et OpenGL
- Beaucoup de veille technologique / documentation
- Apprentissage des bases de la modélisation 3D

# Pistes d'améliorations

- Modéliser un véritable environnement de travail
- Ajout de mouvements de tête pour interagir avec l'environnement virtuel

# Démonstration