Projet CCI-Mobile

Table des matières

I - Contexte

Vous travaillez en tant que technicien développeur junior pour l'ESN Badènia Tech qui vient de remporter le marché de développement d'une application mobile android qui doit participer à la promotion de la chambre de commerce et de l'industrie, particulièrement, les formations proposées. Vous devez faire évoluer l'application existante qui est basique et comporte une seule IHM, la page d'accueil.

II - Analyse des besoins

A- Énoncer le besoin

Afin de rendre l'application de la CCI plus complète, vous devrez rendre l'application plus ergonomique ainsi qu'ajouter des fonctionnalités diverses.

B- Besoin de sécurité

Les informations saisis par les utilisateurs devront être vérifié avant chaque enregistrement et envoie de données.

D- Sprint

1- Sprint 1

Le premier besoin consiste à ajouter un menu nous permettant de scroller une liste composé de toutes les formations disponible.

Pour cela j'ai utilisé un Adapter.

```
formations = controleFormation.getFormations();
Log.i(TAG, msg: "creerListeFormation formations=" + formations);
adapterFormation = new FormationAdapter( c: this, formations);
liste = (ListView) findViewById(R.id.listView);
liste.setAdapter(adapterFormation);
```

```
public FormationAdapter(Context c, List<Formation> ids) {
    mContext = c;
    mFormation = ids;
    inflater = LayoutInflater.from(c);
}

// Return the number of items in the Adapter
@Override
public int getCount() { return mFormation.size(); }

// Return the data item at position
@Override
public Object getItem(int position) { return mFormation.get(position); }

// Will get called to provide the ID that
// is passed to OnItemClickListener.onItemClick()
@Override
public long getItemId(int position) { return position; }

// Return an ImageView for each item referenced by the Adapter
@Override
public View getView(int position, View convertView, ViewGroup parent) {

    TextView textAcronyme;
    TextView textIntitule;
    TextView textDateDebut;
    View view;

    TextView textDebut;
    Formation formation = mFormation.get(position);
    //creations d'une nouvelle view de la liste
    if (convertView == null) {
        view = inflater.inflate(R.layout.view_formation, root: null);
    } else {
        view = convertView;
    }
    textIntitule = (TextView) view.findViewById(R.id.formationIntitule);
    textIntitule.setText(formation.getIntitule());
    textAcronyme = (TextView) view.findViewById(R.id.formationAcronyme);
    textAcronyme.setText("Acronyme: " + formation.getAcronyme());
    textDateDebut = (TextView) view.findViewById(R.id.formationDateDebut);
    textDateDebut.setText("Début: " + formation.getDateDebut());
    return view;
}
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="6sp"
     >


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/formationIntitule"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            style="@style/titre_2"/>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

            android:gravity="center_horizontal"
            android:orientation="horizontal" >
                <TextView
                    android:id="@+id/formationAcronyme"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    style="@style/texteListe" />

                <TextView
                    android:id="@+id/formationDateDebut"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    style="@style/texteListe" />
        </LinearLayout>
    </LinearLayout>
</LinearLayout>
```

2 - Sprint 2

Le second besoin consiste à ajouter une vue qui décrit les formations lorsqu'on les clique dans la liste.

Nous avons donc créé une nouvelle activity ainsi que les api mise à ma disposition

```java
@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
    Log.i(TAG, msg: "onItemClick = " + position);
    Log.i(TAG, msg: "formation = " + formations.get(position));
    controleFormation.setFormation(formations.get(position));
    //Log.i(TAG ,"getFormation : " + controleFormation.getFormation());
    Intent intent = new Intent( packageContext: FormationActivity.this, FormationInfoActivity.class);
    startActivity(intent);
}
```

```java
liste.setOnItemClickListener(this);
```

```java
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_formation_info);
mDbHelper = new DatabaseOpenHelper( context: this);


formation = controleFormation.getFormation();
Log.i(TAG, msg: "getFormation into infoActivity : " + formation);

TextView textIntitule = (TextView) findViewById(R.id.formationIntituleInfo);
textIntitule.setText("Information sur la formation : " + formation.getIntitule());

TextView textDebut = (TextView) findViewById(R.id.formationDateDebutInfo);
textDebut.setText("Débute le : " + Outils.convertStringDate(formation.getDateDebut()));

TextView textDuree = (TextView) findViewById(R.id.formationDureeInfo);
textDuree.setText("Durée totale : " + formation.getDureeMois() + " mois");

ImageView img = (ImageView) findViewById(R.id.imgInfo);
loadImageView(img, url: YoutubeUrlImageBase + formation.getVideoUrl() + YoutubeUrlImageEnd);
img.setOnClickListener(new Button.OnClickListener() {
    @Override
    public void onClick(View v) {
        Log.i(TAG, msg: "Bouton video");
        Intent intent = new Intent( packageContext: FormationInfoActivity.this, VideoActivity.class);
        intent.putExtra( name: "urlVideo", formation.getVideoUrl());
        startActivity(intent);
    }
});
```

```java
TextView textDesc = (TextView) findViewById(R.id.formationDescInfo);
textDesc.setText("Description de la formation : " + formation.getDescription());

TextView textMoreInfo = (TextView) findViewById(R.id.formationMoreInfo);
textMoreInfo.setText("En savoir plus.");


Log.i(TAG, msg: "onCreate");
```

```java
public void enventListenerLink(View view) {
    Log.i(TAG,  msg: "enventListenerLink");
    Intent browserIntent = new Intent(Intent.ACTION_VIEW, Uri.parse(formation.getLink()));
    startActivity(browserIntent);
}

public void loadImageView(ImageView img, String url) {
    //start a background thread for networking
    Log.i( tag: "loadImageView", url);
    new Thread(new Runnable() {
        public void run() {
            try {
                //download the drawable
                final Drawable drawable = Drawable.createFromStream((InputStream) new URL(url).getContent(),  srcName: "src");
                //edit the view in the UI thread
                img.post(new Runnable() {
                    public void run() { img.setImageDrawable(drawable); }
                });
            } catch (IOException e) {
                e.printStackTrace();
                Log.e( tag: "==================erreur test===============",  msg: "test", e);
            }

        }
    }).start();
}
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="6sp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

        android:gravity="center_horizontal"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/formationIntituleInfo"
            style="@style/titre_1"
            android:layout_width="260dp"
            android:layout_height="wrap_content" />

        <ImageView
            android:id="@+id/imgLike"
            android:layout_width="30dp"
            android:layout_height="30dp"
            android:layout_gravity="center_vertical"
            android:contentDescription="Like" />
    </LinearLayout>


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

        android:gravity="center_horizontal"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/formationDateDebutInfo"
            style="@style/texteListe"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />

        <TextView
            android:id="@+id/formationDureeInfo"
            style="@style/texteListe"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <ImageView
        android:id="@+id/imgInfo"
        android:layout_width="match_parent"
        android:layout_height="150dp"
        android:layout_gravity="center_horizontal"
        android:clickable="true"
        android:focusable="true"
        tools:ignore="SpeakableTextPresentCheck" />

    <TextView
        android:id="@+id/formationDescInfo"
        style="@style/texteListe"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />


    <TextView
        android:id="@+id/formationMoreInfo"
        style="@style/boutonLink"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="enventListenerLink" />
</LinearLayout>
```

3 – Sprint 3

Le troisième besoin consiste à pouvoir poster des messages au staff de l'équipe. Pour cela j'ai créé une nouvelle activity et un nouvel accès par le menu. Ensuite j'ai mise en place des vérification d'entrée.

```java
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_contact);
    controleMessage = ControleMessage.getInstance();

    EditText inputFirstname = (EditText) findViewById(R.id.firstnameForm);
    EditText inputLastname = (EditText) findViewById(R.id.lastnameForm);
    EditText inputEmail = (EditText) findViewById(R.id.emailForm);
    EditText inputContent = (EditText) findViewById(R.id.contentForm);
    Button buttonSubmit = (Button) findViewById(R.id.boutonSubmit);
    TextView popup = (TextView) findViewById(R.id.popup);
    buttonSubmit.setOnClickListener(
            new View.OnClickListener()
            {
                public void onClick(View view)
                {
                    String nom = inputLastname.getText().toString();
                    String prenom = inputFirstname.getText().toString();
                    String email = inputEmail.getText().toString();
                    String content = inputContent.getText().toString();
                    Log.i(TAG, inputFirstname.getText().toString());
                    Log.i(TAG, inputLastname.getText().toString());
                    Log.i(TAG, inputEmail.getText().toString());
                    Log.i(TAG, inputContent.getText().toString());
                    Log.i(TAG,  msg: "form valid ? "+ controleMessage.isValidForm(nom, prenom,email,content));
                    popup.setText(null);
                    if (controleMessage.isValidForm(nom, prenom,email,content)){
                        controleMessage.createMessage(nom, prenom, email,  formation: null ,content);
                        inputLastname.setText(null);
                        inputFirstname.setText(null);
                        inputEmail.setText(null);
                        inputContent.setText(null);
                        popup.setText("Message envoyé");
                    }else {
                        popup.setText("Veuillez saisir des informations valides");
                    }
                }
            });
}
```

```java
public boolean isValidEmail(String string){
    final String EMAIL_PATTERN = "^[_A-Za-z0-9-\\+]+(\\.[_A-Za-z0-9-]+)*@[A-Za-z0-9-]+(\\.[A-Za-z0-9]+)*(\\.[A-Za-z]{2,})$";
    Pattern pattern = Pattern.compile(EMAIL_PATTERN);
    Matcher matcher = pattern.matcher(string);
    return matcher.matches();
}
public boolean isOnlyCharater(String string){
    final String LETTER_PATTERN = "[a-zA-Zéèçàêëî]+";
    Pattern pattern = Pattern.compile(LETTER_PATTERN);
    Matcher matcher = pattern.matcher(string);
    return matcher.matches();
}
public boolean isNullOrEmpty(String string){
    if(string == null) return true;
    else if(string.isEmpty()) return true;
    else return false;
    //return TextUtils.isEmpty(string);
}

public boolean isNumeric(String string){
    //return TextUtils.isDigitsOnly(string);
    final String LETTER_PATTERN = "[0-9.]+";
    Pattern pattern = Pattern.compile(LETTER_PATTERN);
    Matcher matcher = pattern.matcher(string);
    return matcher.matches();
}
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:baselineAligned="false"
    android:orientation="vertical">

    <TextView
        style="@style/titre_1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Nous Contacter" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:orientation="horizontal">

        <EditText
            android:id="@+id/lastnameForm"
            android:layout_width="150dp"
            android:layout_height="match_parent"
            android:hint="Nom"
            android:inputType="text" />

        <EditText
            android:id="@+id/firstnameForm"
            android:layout_width="150dp"
            android:layout_height="match_parent"
            android:hint="Prénom"
            android:inputType="text" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:orientation="horizontal">

        <EditText
            android:id="@+id/emailForm"
            android:layout_width="300dp"
            android:layout_height="50dp"
            android:hint="Adresse Mail"
            android:inputType="textEmailAddress"
            android:paddingTop="10dp" />
    </LinearLayout>

    <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="200dp"
        android:fillViewport="false"
        android:orientation="vertical"
        android:padding="10dp">


        <EditText
            android:id="@+id/contentForm"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:hint="Contenu du message"
            android:inputType="textMultiLine"
            android:paddingTop="10dp" />
    </ScrollView>
    <TextView
        style="@style/texte"
        android:id="@+id/popup"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="" />
    <Button
        android:id="@+id/boutonSubmit"
        style="@style/bouton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Envoyer" />
</LinearLayout>
```

4 – Sprint 4

Le dernier besoin consiste a pouvoir enregistrer de façon local des formations en favoris.

Pour cela je créé une base de données dans le device, et je change le statut des formations mise en favori. Je créé aussi une activity afin de pouvoir consulter les favoris enregistrés.

```java
ImageView imgLike = (ImageView) findViewById(R.id.imgLike);
fav = controleFavoris.getFavoris(formation.getAcronyme(), context: this);
if (fav) {
    imgLike.setImageResource(R.drawable.favori);
} else {
    imgLike.setImageResource(R.drawable.favorino);
}
imgLike.setOnClickListener(new Button.OnClickListener() {
    @Override
    public void onClick(View v) {
        Log.i(TAG, msg: "Bouton fav");
        fav = !fav;
        controleFavoris.setFavoris(fav, formation.getAcronyme(), context: FormationInfoActivity.this);
        if (fav) {
            imgLike.setImageResource(R.drawable.favori);
        } else {
            imgLike.setImageResource(R.drawable.favorino);
        }
    }
});
```

```java
public final static String TABLE_NAME = "formation";
public final static String FORMATION_ACRONYME = "acronyme";
public final static String FORMATION_FAVORIS = "favoris";

final static String _ID = "_id";

final static String[] columns = {_ID, FORMATION_ACRONYME, FORMATION_FAVORIS};

final private static String CREATE_CMD =
        "CREATE TABLE formation (" + _ID
                + " INTEGER PRIMARY KEY AUTOINCREMENT, "
                + FORMATION_ACRONYME + " TEXT NOT NULL, "
                + FORMATION_FAVORIS + " BOOLEAN NOT NULL )";

final private static String NAME = "cci_db";
final private static Integer VERSION = 1;
final private Context mContext;

public DatabaseOpenHelper(Context context) {
    super(context, NAME, factory: null, VERSION);
    this.mContext = context;
}

@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL(CREATE_CMD);

}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    // N/A
}

void deleteDatabase() { mContext.deleteDatabase(NAME); }

public void clearAll() {

    this.getWritableDatabase().delete(DatabaseOpenHelper.TABLE_NAME, whereClause: null, whereArgs: null);

}
```

```java
public void setFavoris(Boolean favoris, String acronyme, Context context) {
    ContentValues values = new ContentValues();
    values.put("favoris", favoris);
    mDbHelper = new DatabaseOpenHelper(context);
    mDbHelper.getWritableDatabase().update( table: "formation", values, whereClause: "acronyme = ?", new String[]{acronyme});
}

public Boolean getFavoris(String acronyme, Context context) {
    boolean fav = false;
    mDbHelper = new DatabaseOpenHelper(context);
    Cursor cursor = mDbHelper.getReadableDatabase().query( table: "formation", columns: null, selection: "formation.acronyme = ?", new String[]{acronyme}, groupBy: null, having: null,
    if (cursor != null) {
        cursor.moveToFirst();
        int n = cursor.getColumnIndex(DatabaseOpenHelper.FORMATION_FAVORIS);
        fav = cursor.getInt(n) > 0;
    }
    return fav;
}

public void setFavorisDevice(Context context, List<Formation> formations) {
    ContentValues values = new ContentValues();
    mDbHelper = new DatabaseOpenHelper(context);
    Cursor cursor = mDbHelper.getReadableDatabase().query( table: "formation", columns: null, selection: null, selectionArgs: null, groupBy: null, having: null, orderBy: null);
    cursor.moveToFirst();
    int n = cursor.getCount();
    if (n == 0) {
        Log.i( tag: "repo", msg: "pas de bdd local");
        for (Formation formation : formations) {
            values.put(DatabaseOpenHelper.FORMATION_ACRONYME, formation.getAcronyme());
            values.put(DatabaseOpenHelper.FORMATION_FAVORIS, false);
            mDbHelper.getWritableDatabase().insert(DatabaseOpenHelper.TABLE_NAME, nullColumnHack: null, values);
            values.clear();
        }
    }
}
```

```java
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_favoris);
    Log.i(TAG, msg: "OnCreate");
    controleFormation = ControleFormation.getInstance();
    controleFavoris = ControleFavoris.getInstance();
    acroList = controleFavoris.getFavorisAcronyme( context: this);
    if (acroList != null) {
        Log.i(TAG, msg: "longueur = " + acroList.size());
        creerListeFavorisFormation();
    } else {
        Log.i(TAG, msg: "accroList null");
    }
}

private void creerListeFavorisFormation() {
    Log.i(TAG, msg: "creerListeFormation formations=" + acroList);
    ArrayAdapter<String> adapter = new ArrayAdapter<~>( context: this, android.R.layout.activity_list_item, android.R.id.text1, acroList);
    liste = (ListView) findViewById(R.id.listViewFav);
    liste.setAdapter(adapter);
    liste.setOnItemClickListener(this);
}

@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
    Log.i(TAG, msg: "onItemClick = " + position);
    Log.i(TAG, msg: "acro = " + acroList.get(position));
    controleFormation.setFormation(controleFormation.getFormationByAcronyme(acroList.get(position)));
    Log.i(TAG, msg: "formation du ctrl = " + controleFormation.getFormation());
    Intent intent = new Intent( packageContext: FavorisActivity.this, FormationInfoActivity.class);
    startActivity(intent);

}
```

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:baselineAligned="false"
    android:orientation="vertical" >
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Liste des favoris"
        style="@style/titre_1" />
    <ListView
        android:id="@+id/listViewFav"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        />
</LinearLayout>
```

E – Bilan de la production

Les versions on était fournis à temps et tous les objectifs ont été rempli. Néanmoins le code peut être plus optimisé. Durant cette réalisation j'ai pu mettre en œuvre mes compétences en Java et SQL. J'ai aussi pu me familiarisé avec le framework Android.

III - Compétences validées

J'ai pu validé au travers de cette réalisation 3 compétences.

- Développer la présence en ligne de l'organisation

- Travailler en mode projet

- Mettre à disposition des utilisateurs un service informatique

Pour plus d'informations veuillez consulter ce document