



App ins Grüne

Android Praktikum Lisa Görtz

Multimedia Praktikum Cara Emberger

SoSe 2020

Quizfragen:

minigames_choose_answer.xml

Eberesche:

Blüten und Früchte:

2. Wie werden die Früchte der Eberesche genannt?

A: Eberbeeren

B: Vogelbeeren*

C: Johannisbeeren

3. Wonach riechen die rund 1cm großen Blüten?

A: altem Fisch*

B: Zitrone

C: Hustenbonbons

Rinde und Holz:

3. Was hat man aus dem Holz der Eberesche machen?

A: Pfeifenköpfe

B: Wagenräder*

C: Essstäbchen

```
<minigame_chooseAnswer>
  <id>97</id>
  <name>Quiz</name>
  <description>Wonach riechen die rund 1 cm großen Blüten?</description>
  <selection>
    <option>
      <option_type>Text</option_type>
      <option_content>A: altem Fisch</option_content>
      <option_right>true</option_right>
    </option>
    <option>
      <option_type>Text</option_type>
      <option_content>B: Zitrone</option_content>
      <option_right>false</option_right>
    </option>
    <option>
      <option_type>Text</option_type>
      <option_content>C: Hustenbonbons</option_content>
      <option_right>false</option_right>
    </option>
  </selection>
</minigame_chooseAnswer> <!-- Eberesche -->
```

Texte für Quizfragen

- Quiz: minigames_chooseanswer.xml
- Schreibarbeit
- Copy & Paste
- Immer 4 Fragen: True / False -> aber: eigentlich Datenbank
- Selbes Vorgehen mit Puzzle, etc.

Nach Frage 1 kommt Frage 2...

Automatische Weiterleitung

DataManager.java

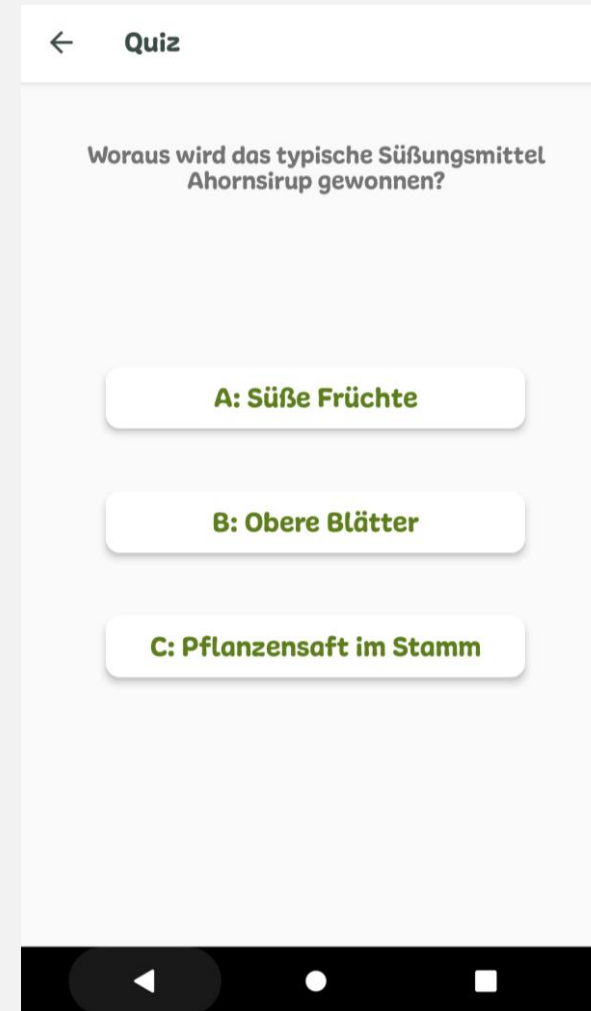
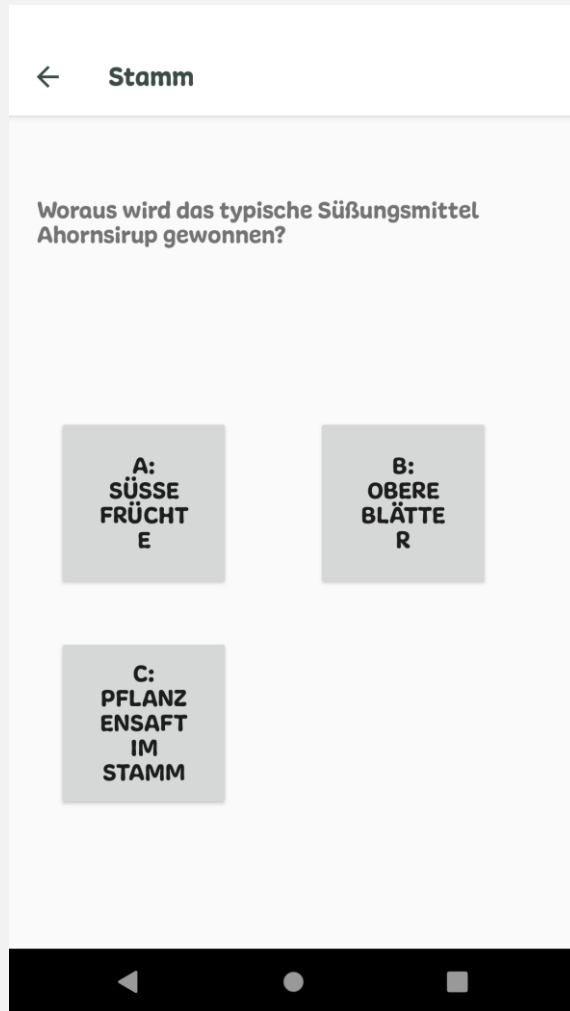
```
public int getNextQuizID(){  
    return DataManager.getInstance(getApplicationContext()).GetNextQuiz(gameContent.uid).uid;  
}
```

```
public Minigame_Base GetNextQuiz(int id) {  
    if (miniGames == null) return null;  
    for (int i=0; i<miniGames.size(); i++){  
        if (miniGames.get(i).uid == id){  
            return miniGames.get(i+10);  
        }  
    }  
    return null;  
}
```

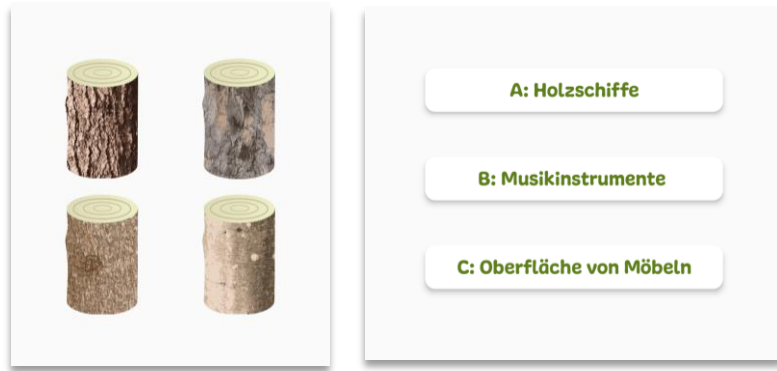
```
private void showNextQuestion(){  
    Intent nextQuestion = new Intent(getApplicationContext(), GameActivity_ChooseAnswer.class);  
    nextQuestion.putExtra(name: "TreeId", treeId);  
    nextQuestion.putExtra(name: "Category", parentCategory);  
    nextQuestion.putExtra(name: "GameId", getNextQuizID());  
    popupWindow.dismiss();  
    finish(); // Removes the current quiz activity from the stack  
    startActivity(nextQuestion);  
}
```

GameActivity_ChooseAnswer.java

Button Design

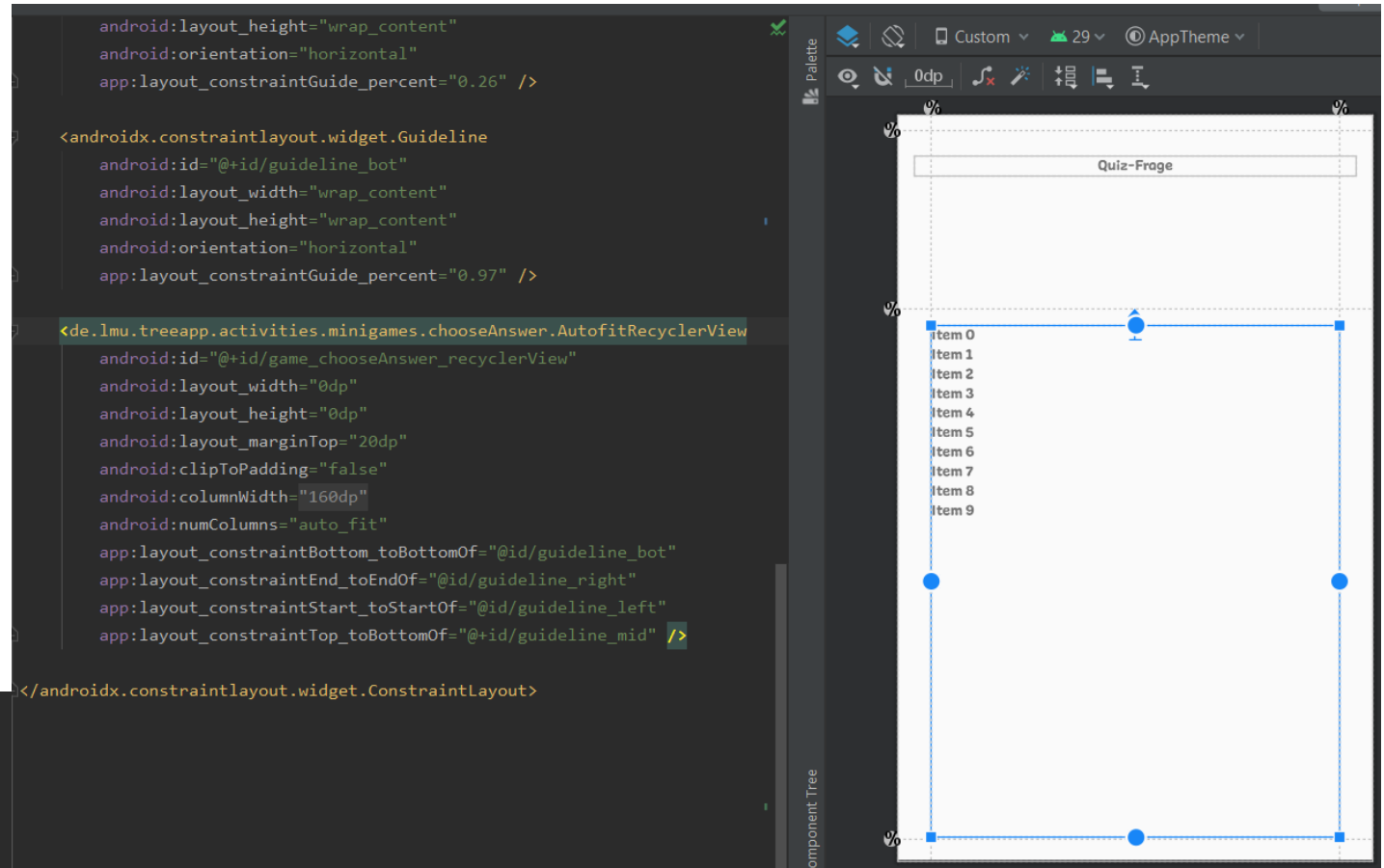


Spaltenanzahl



AutofitRecyclerView.java

```
@Override
protected void onMeasure(int widthSpec, int heightSpec) {
    super.onMeasure(widthSpec, heightSpec);
    if (columnWidth > 0) {
        int spanCount;
        if(ChooseAnswer_Options_RecyclerViewAdapter.isImage(ChooseAnswer_Options_RecyclerViewAdapter.options)){
            spanCount = 2;
        }else{
            spanCount = 1;
        }
        manager.setSpanCount(spanCount);
    }
}
```



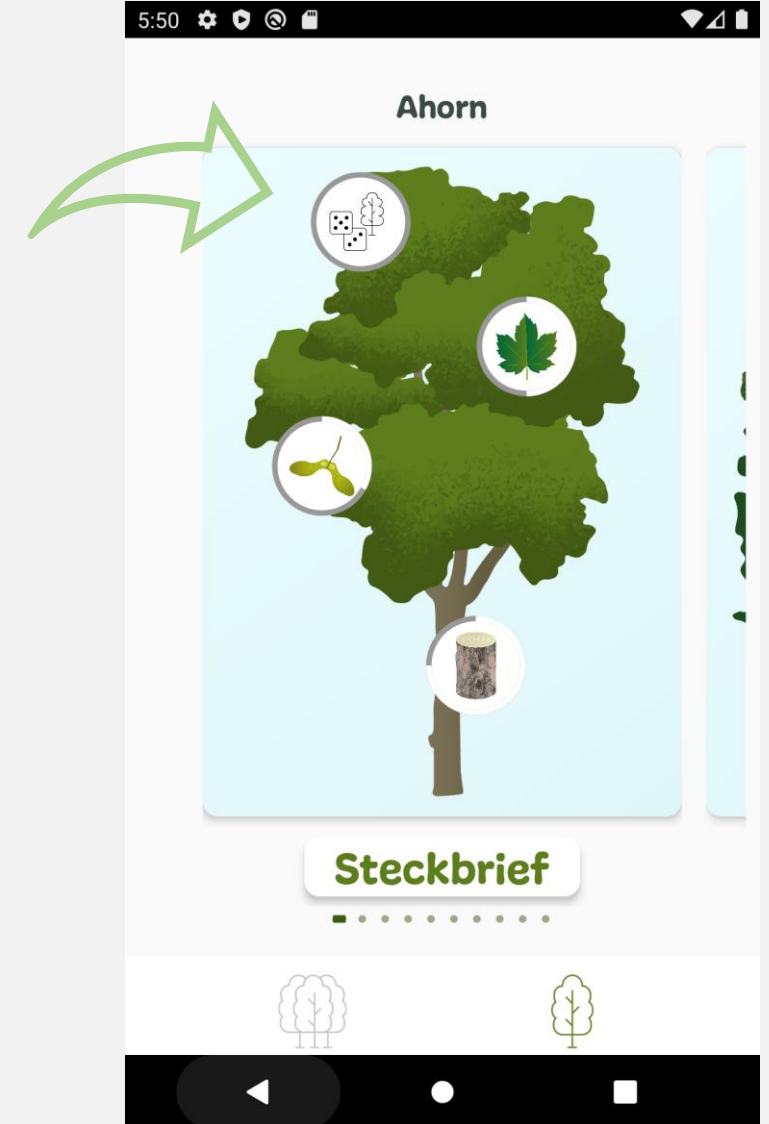
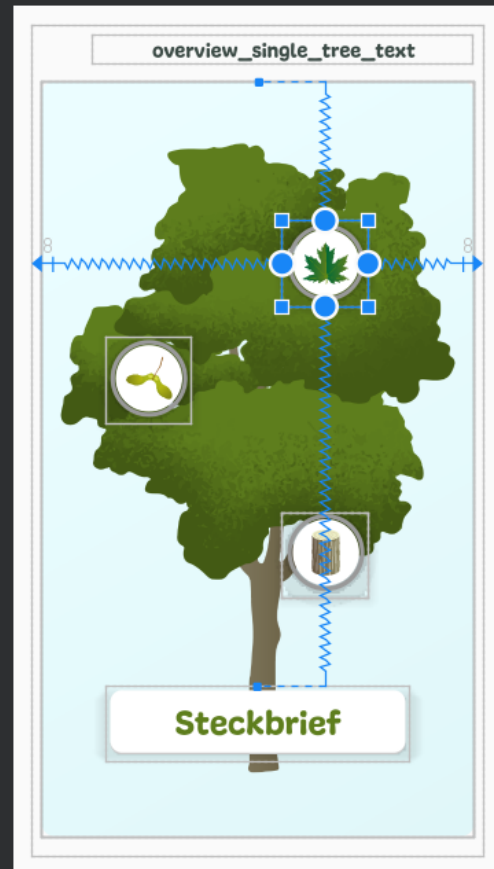
activity_game_choose_answer.xml

Button: Allgemeine Spiele

```
app:layout_constraintVertical_bias="0.873" />

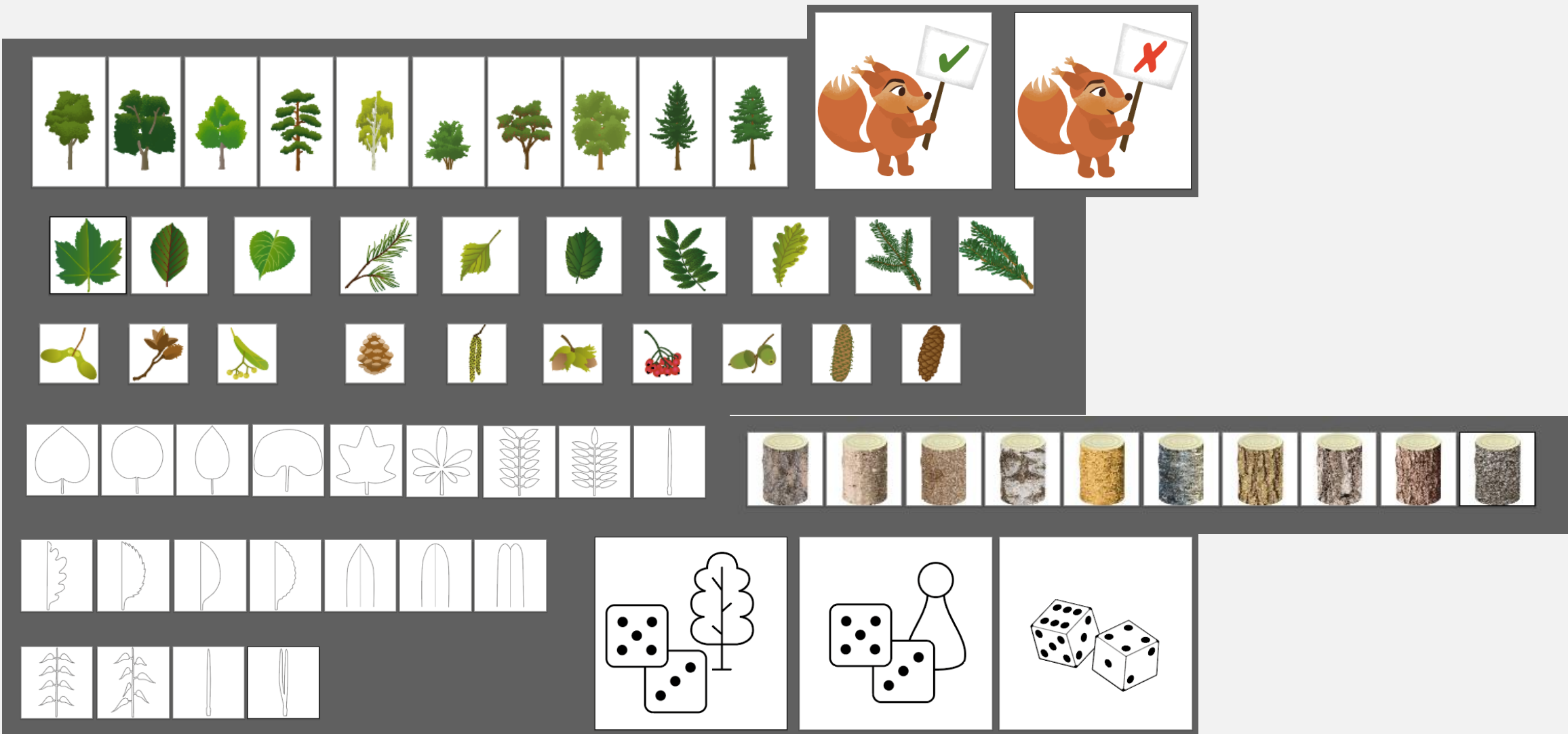
<ImageButton
    android:id="@+id/detail_single_tree_leafButton"
    android:layout_width="72dp"
    android:layout_height="72dp"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginBottom="8dp"
    android:background="@drawable/ic_button_background"
    android:elevation="4dp"
    android:padding="16dp"
    android:scaleType="fitCenter"
    android:contentDescription="TODO"
    app:layout_constraintBottom_toTopOf="@+id/detail_single_tree_profileButton"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.697"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="@+id/detail_single_tree_image"
    app:layout_constraintVertical_bias="0.259"
    tools:srcCompat="@drawable/ic_ahorn_blatt" />

<ImageButton
    android:id="@+id/detail_single_tree_fruitButton"
    android:layout_width="72dp"
    android:layout_height="72dp"
```



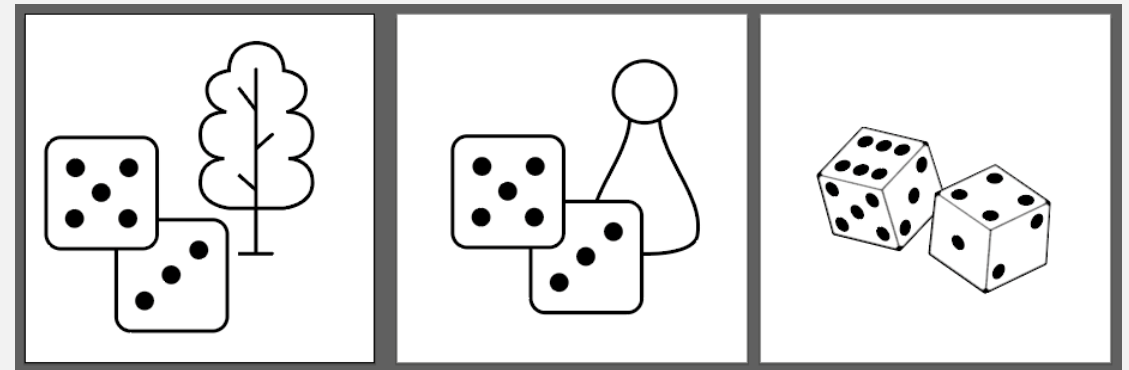
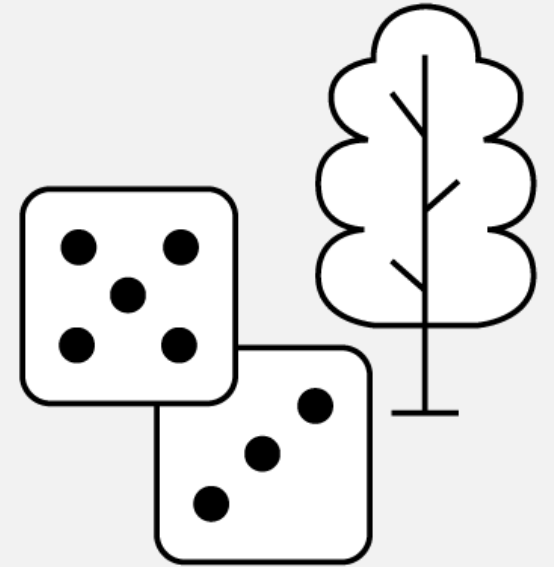
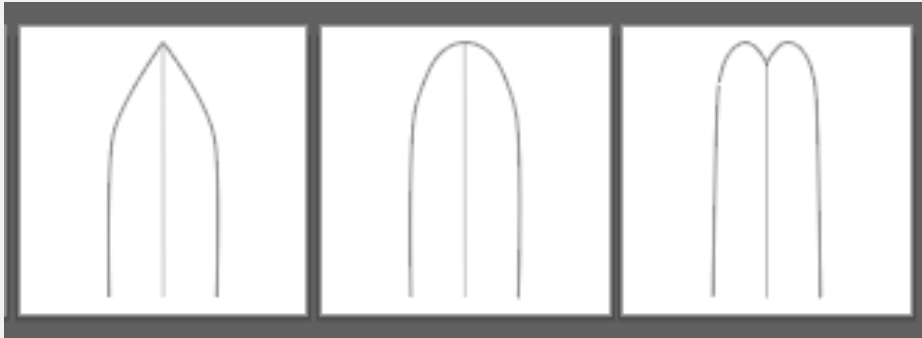
fragment_detail_single_tree.xml

Icons

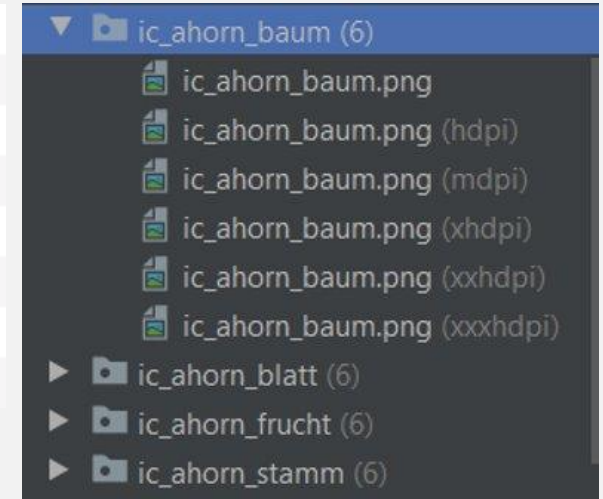
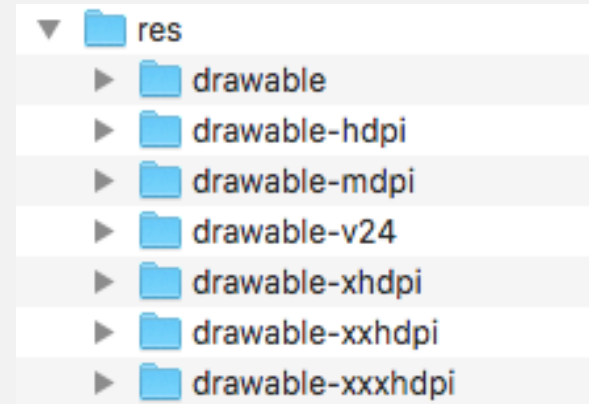
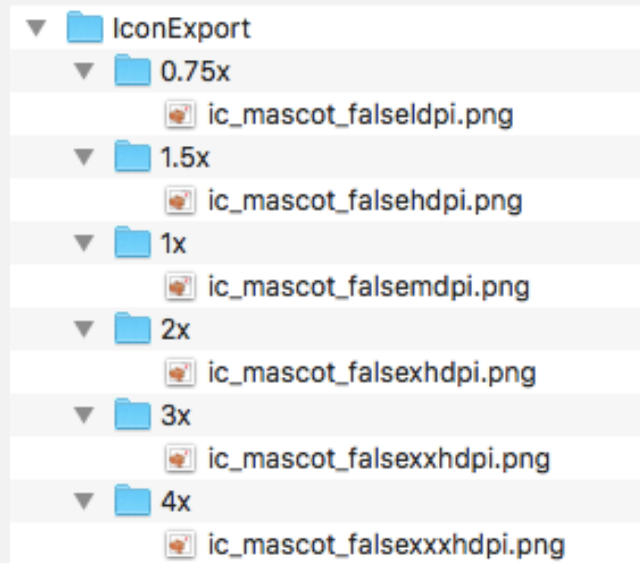
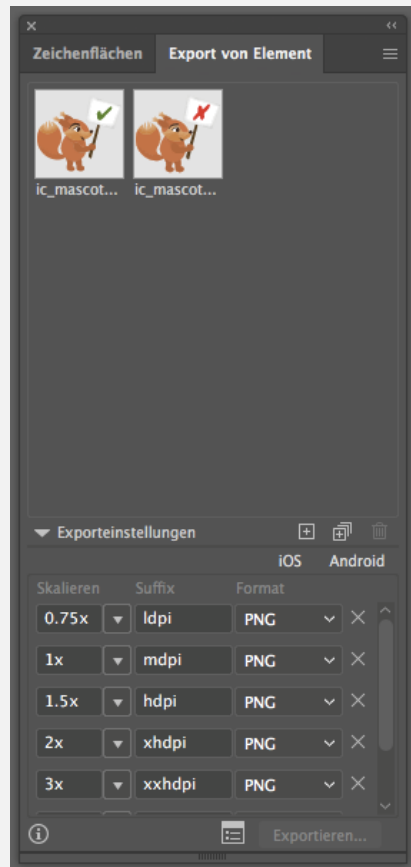


Neue Icons

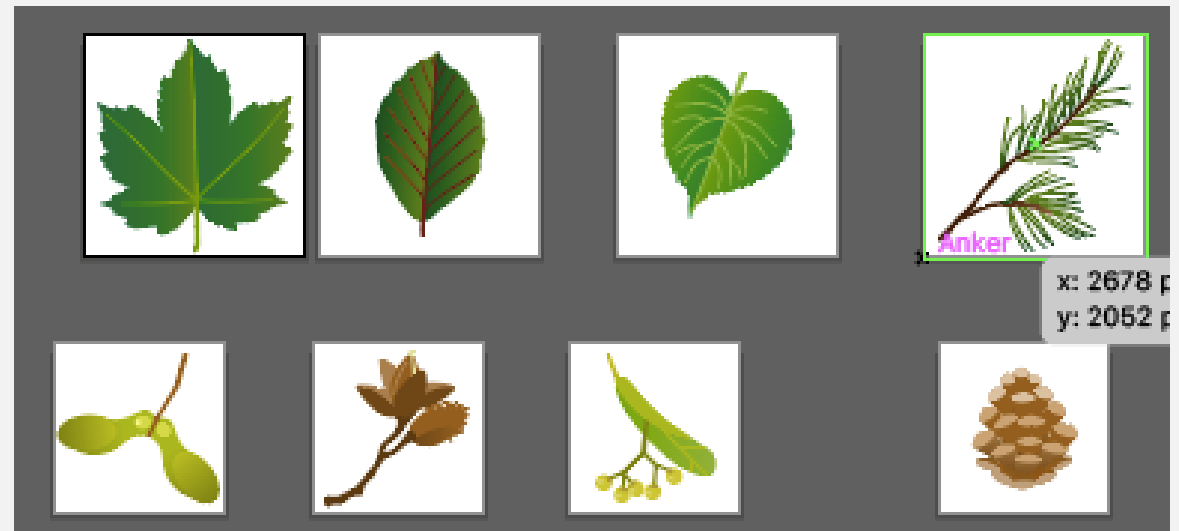
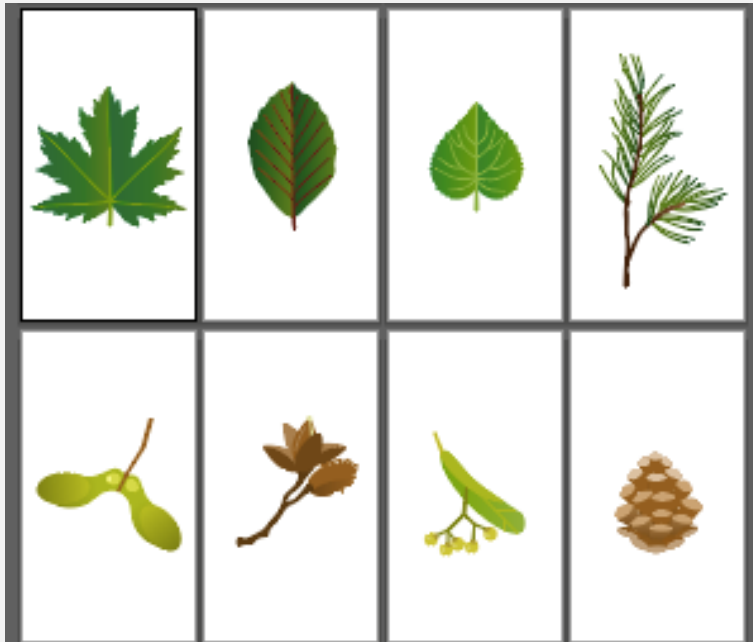
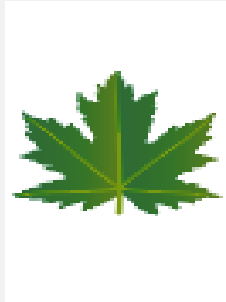
- Unvorhergesehene Icons



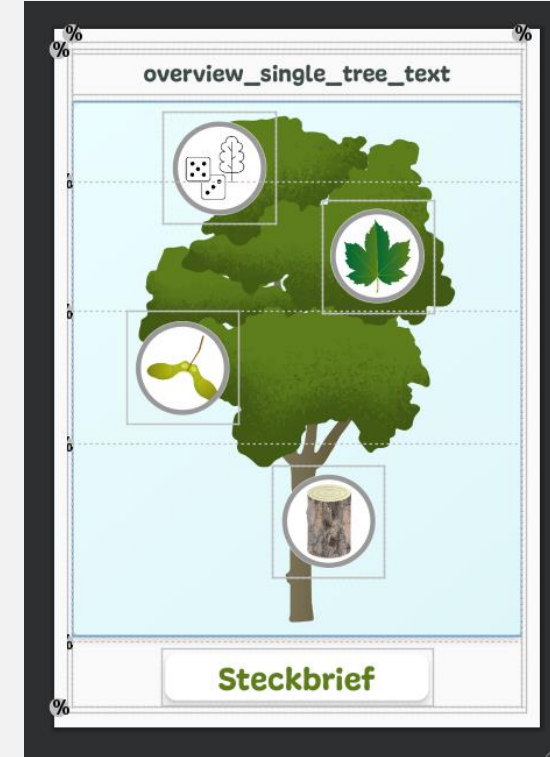
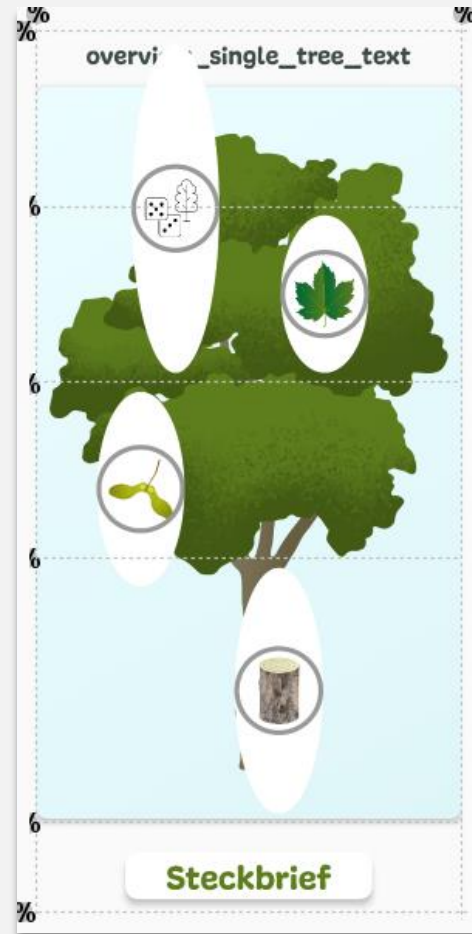
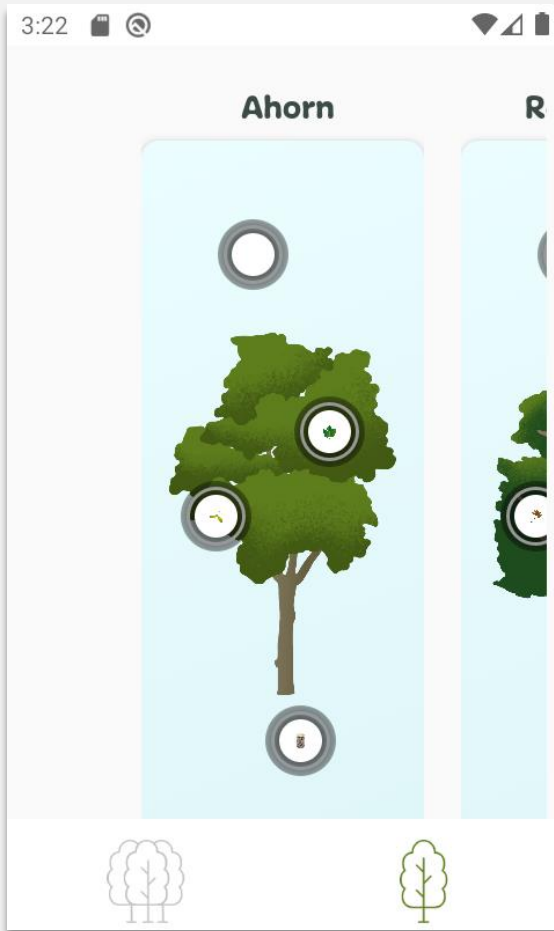
Icon Import



Icons



Skalierung der Icons



- Erneuter Import aller Icons im quadratischen Format
- Constraint Layout
- Hilfslinien (Guidelines)
- Icon ratio 1:1

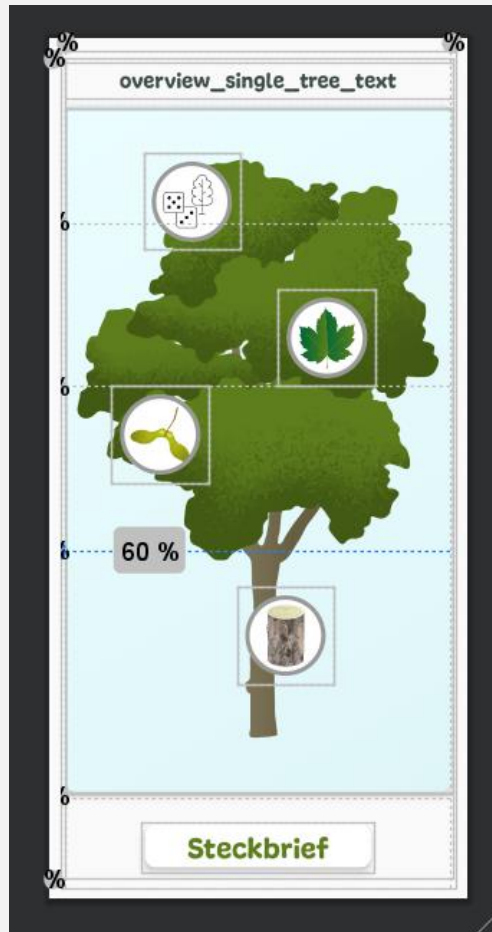
```
<androidx.constraintlayout.widget.ConstraintLayout
    android:id="@+id/constraintLayout_DetailSingleTree"
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintBottom_toTopOf="@+id/guideline_bot"
    app:layout_constraintEnd_toStartOf="@+id/guideline_right"
    app:layout_constraintStart_toStartOf="@+id/guideline_left"
    app:layout_constraintTop_toBottomOf="@+id/guideline_top">
```

```
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guidelineInsideView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.2" />

<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guidelineInsideView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.4" />

<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guidelineInsideView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.6" />

<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guidelineInsideView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.9" />
```



fragment_detail_single_tree.xml

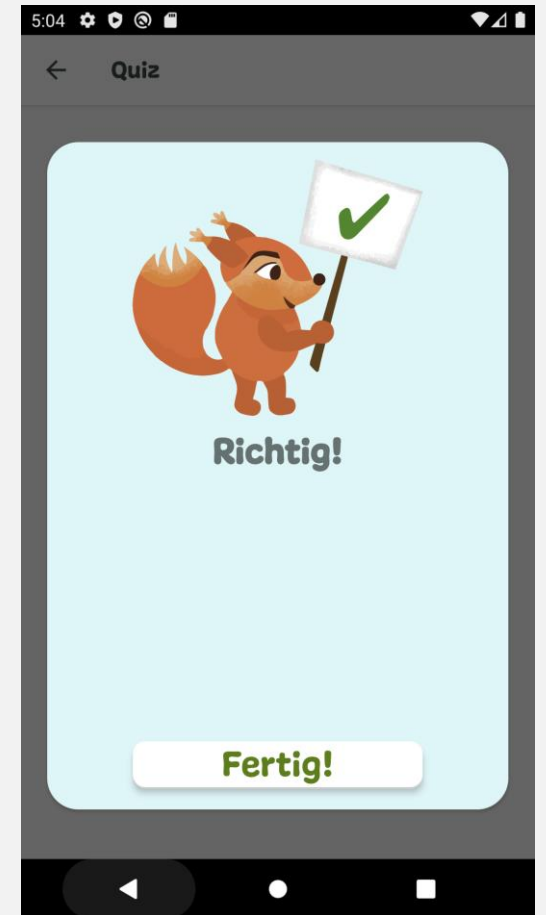
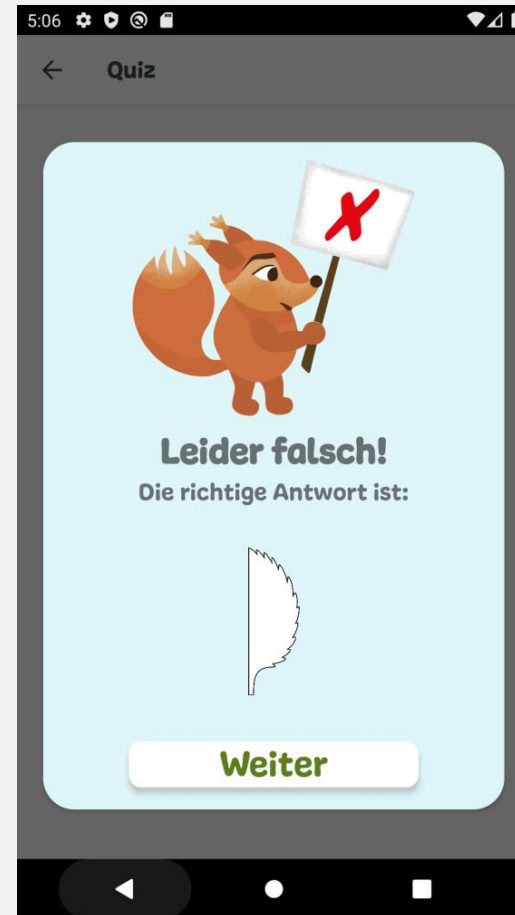
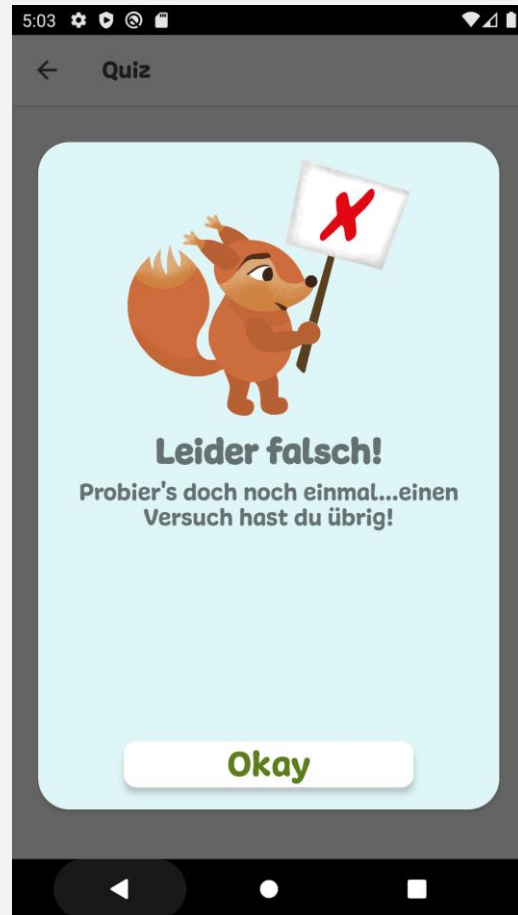
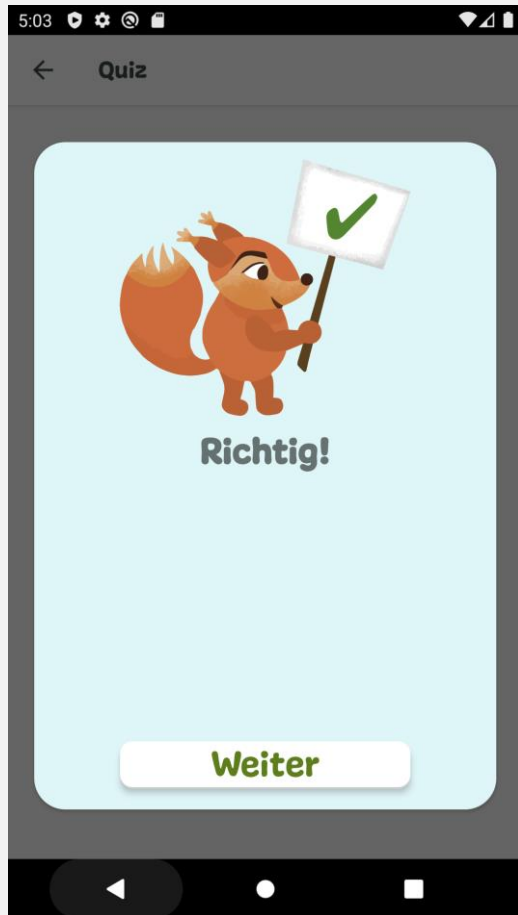
Maskottchen



Maskottchen & Pop-Up



Pop-Up: Design




Pop-Up: Code

GameActivity_ChooseAnswer.java

```
@Override
public void optionClicked(AnswerOption option){
    //quizIDs.add(gameContent.uid);

    if(option.right) {
        showPositivePopup();
    } else {
        showAnswer++;
        showNegativePopup(option);
    }
}
```



```
// Create popup which gives feedback to the user's answer
private void showPositivePopup() {
    popupWindow.setContentView(R.layout.popup_quiz_positive);
    btnAccept = popupWindow.findViewById(R.id.forward_next_game_positive);
    popupTitle = popupWindow.findViewById(R.id.popup_positive_title);

    // Checks if the current activity is last one
    if(current>1) {
        btnAccept.setText("Weiter");
    } else btnAccept.setText("Fertig!");
}
```

```
// Close the popup to repeat the question or finish the game and go back to the overview
btnAccept.setOnClickListener((v) → {
    quizIDs.add(gameContent.uid);
    if(current>1){
        showNextQuestion();
        current--;
    } else {
        popupWindow.dismiss();
        onQuizSuccess(quizIDs);
    }
});
```

Antwortmöglichkeiten: Bild oder Text?

ChooseAnswer_Options_RecyclerViewAdapter.java

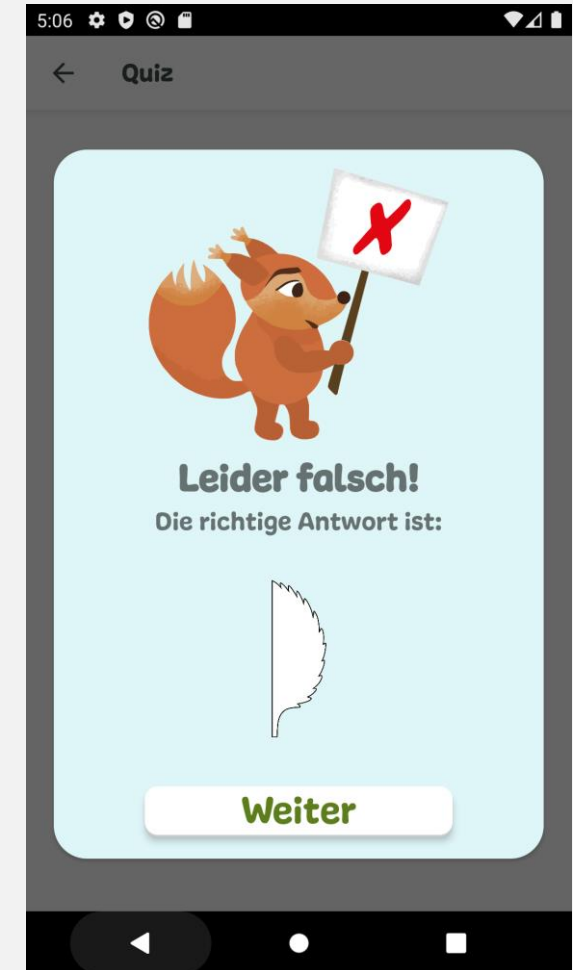
```
@Override
public void onBindViewHolder(ChooseAnswer_Options_RecyclerViewAdapter.ViewHolder holder, final int position) {
    final AnswerOption option = options.get(position);

    if(option.type == AnswerOption.OptionTypes.text){
        holder.button_txt.setVisibility(View.VISIBLE);
        holder.button_txt.setText(option.content);
        if(option.right){
            resultText = holder.button_txt.getText().toString();
            MainActivity_ChoseAnswer.resultText = resultText;
        }
    }else if(option.type == AnswerOption.OptionTypes.image){
        holder.button_img.setVisibility(View.VISIBLE);
        int imageId = context.getResources().getIdentifier(option.content, defType: "drawable", context.getPackageName());

        if(option.right){
            resultImageId = imageId;
            MainActivity_ChoseAnswer.resultImage = resultImageId;
        }
        holder.button_img.setBackgroundResource(imageId);
        holder.button_img.setText("");

        holder.button_img.setOnClickListener((arg0) -> {
            mOnClickListener.optionClicked(option);
        });

        holder.button_txt.setOnClickListener((arg0) -> {
            mOnClickListener.optionClicked(option);
        });
    }
}
```



Spiel-Fortschritt speichern

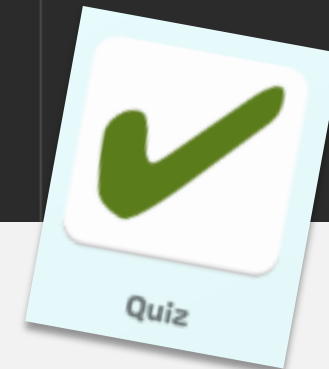
GameActivity_Base.java

```
// Mark all quiz questions as completed and close the quiz
protected void onQuizSuccess(ArrayList<Integer> quizIDs){
    if (quizIDs == null || quizIDs.isEmpty()) return;
    System.out.println(quizIDs);

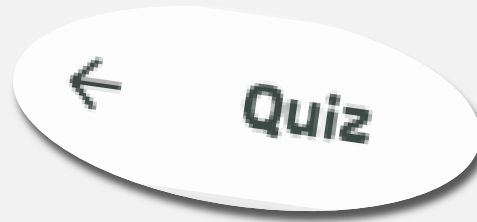
    for(int i=0; i<quizIDs.size(); i++){
        DataManager.getInstance(getApplicationContext()).GameCompleted(parentCategory, quizIDs.get(i), parentTree);
    }
    quizIDs.clear();
    System.out.println(quizIDs);

    Intent intent = new Intent(getApplicationContext(), GameSelectionActivity.class);
    intent.putExtra( name: "TreeId", treeId);
    intent.putExtra( name: "Category", parentCategory);
    finish(); // Removes the last quiz activity from the stack
    startActivity(intent);
}
```

```
I/System.out: [0, 10, 20, 30]
I/System.out: []
```



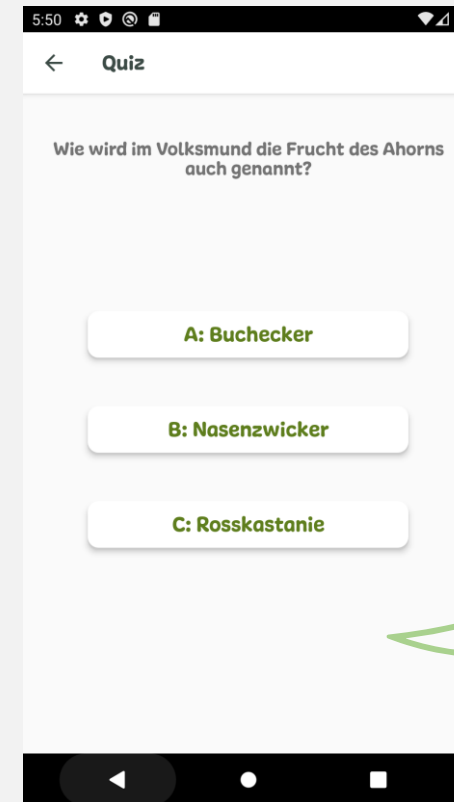
Zurück klicken



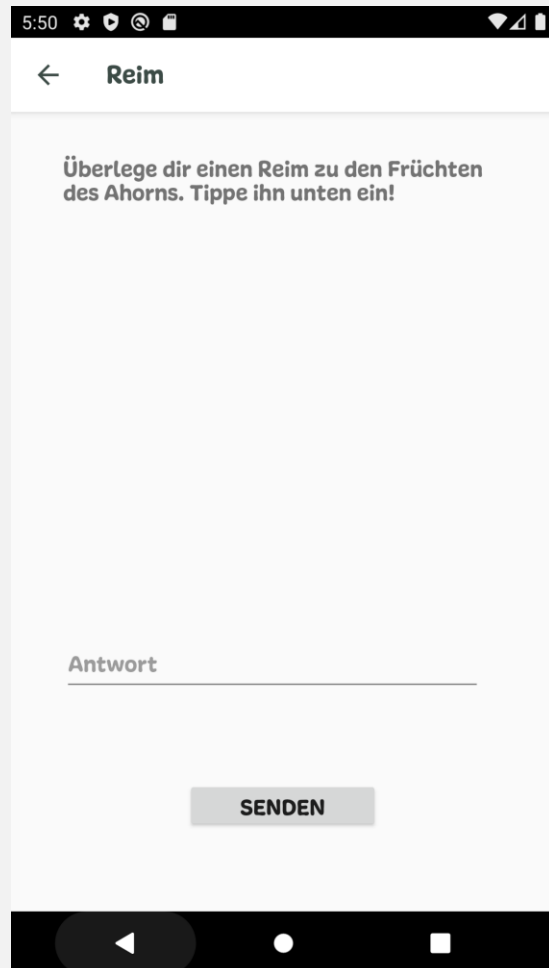
```
// Remove the current activity from the stack to go back to the previous one
@Override
public boolean onSupportNavigateUp(){
    if(gameContent.type.name().equalsIgnoreCase( anotherString: "ChooseAnswer")){
        GameActivity_ChooseAnswer.quizIDs.clear();
    }
    finish();
    return true;
}

// Android hardware back button is pressed
@Override
public void onBackPressed(){
    if(gameContent.type.name().equalsIgnoreCase( anotherString: "ChooseAnswer")) {
        GameActivity_ChooseAnswer.quizIDs.clear();
    }
    finish();
    super.onBackPressed();
}
```

GameActivity_Base.java



Minispiel: Texteingabe



activity_game_input_string.xml

```
<Button
    android:id="@+id/game_inputString_sendButton"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:layout_marginBottom="20dp"
    android:background="@drawable/button_background"
    android:elevation="4dp"
    android:fontFamily="@font/freude"
    android:paddingTop="8dp"
    android:paddingBottom="4dp"
    android:stateListAnimator="@null"
    android:text="@string/game_input_string_send_button_placeholder_text"
    android:textAppearance="@style/TextAppearance.Compat.Notification.Title.Media"
    android:textColor="@color/forest"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/guideline3"
    app:layout_constraintVertical_bias="0.8"
    app:layout_constraintWidth_percent="0.6" />
```

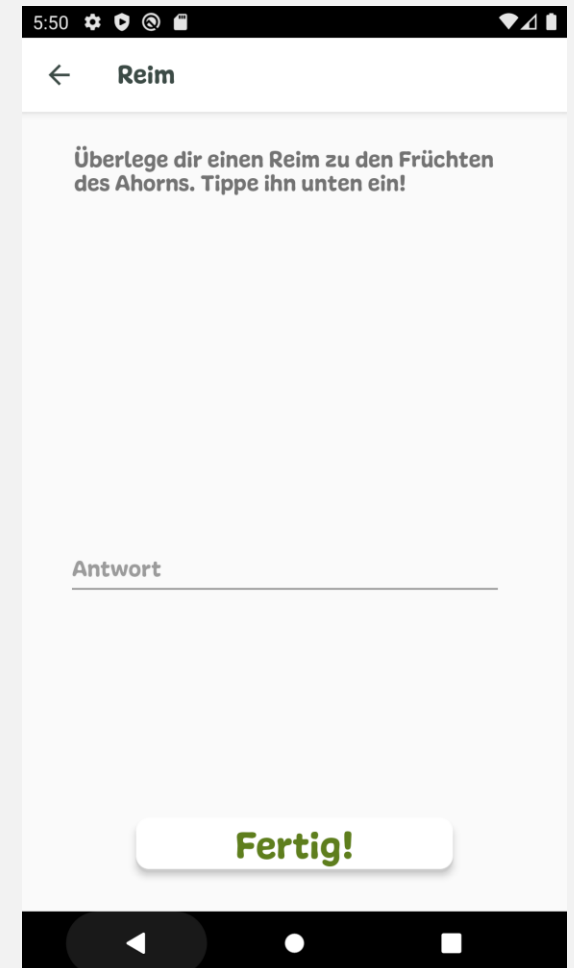
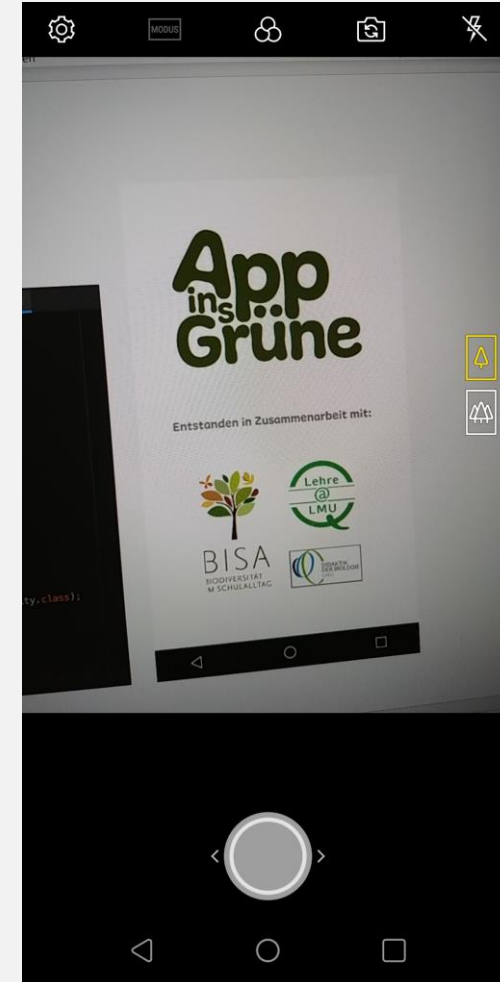
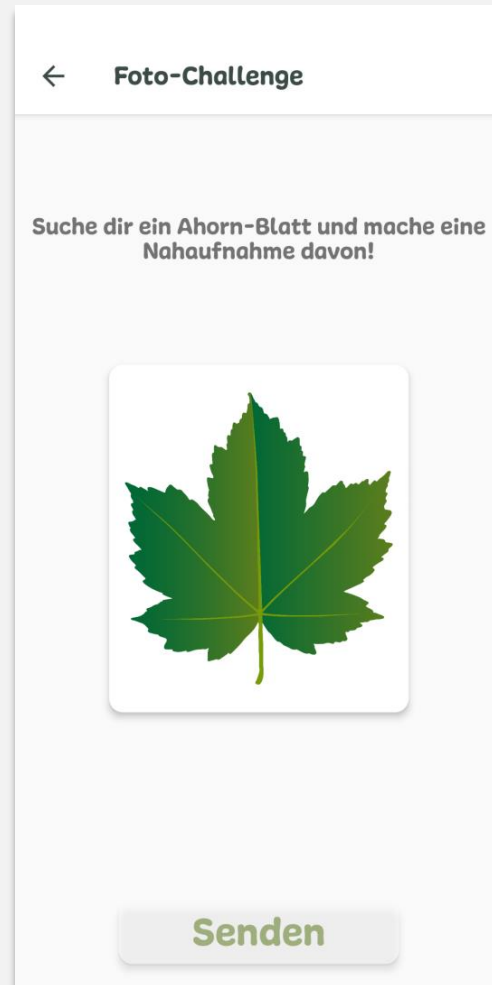
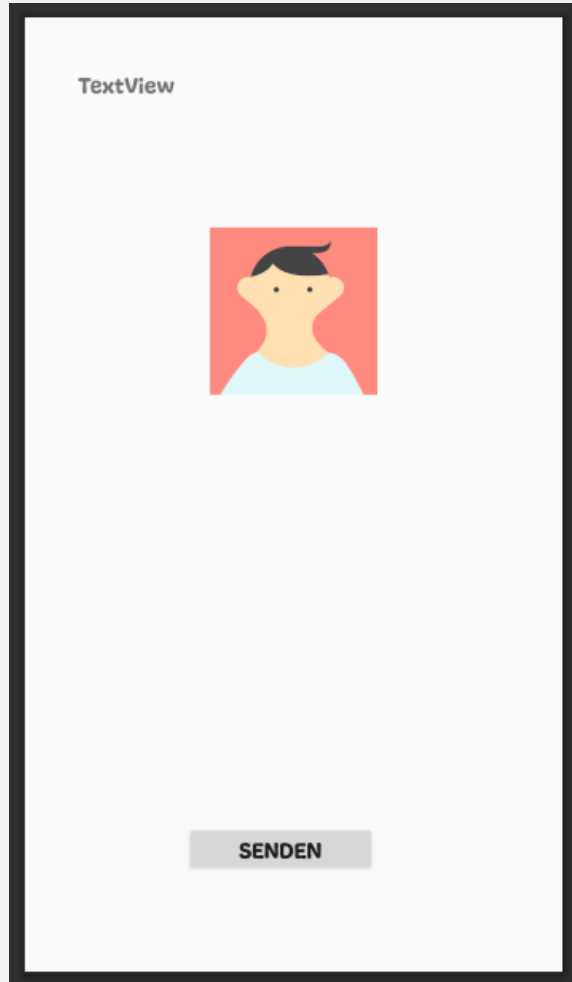


Foto-Challenge



← Foto-Challenge

Suche dir ein Ahorn-Blatt und mache eine Nahaufnahme davon!



Senden

← Foto-Challenge

Suche dir ein Ahorn-Blatt und mache eine Nahaufnahme davon!



Senden



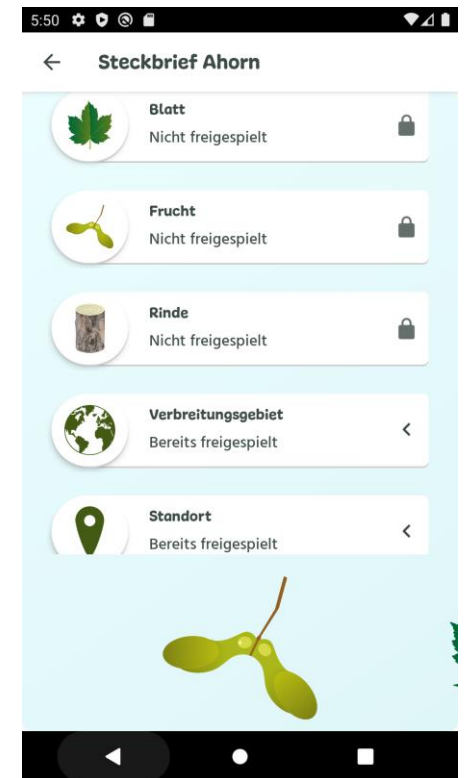
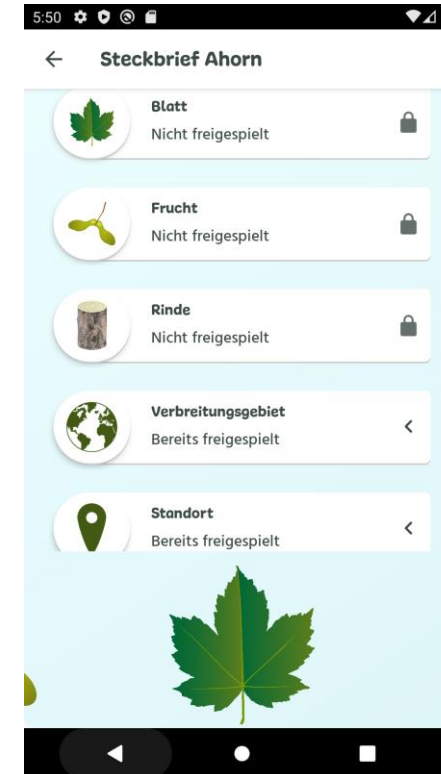
Foto-Galerie

WantedPosterDetailsActivity.java

```
// ViewPager2
viewPager2 = findViewById(R.id.viewPagerImageSlider);
List<SliderItem> sliderItems = new ArrayList<>();
sliderItems.add(new SliderItem(R.drawable.ic_ahorn_frucht));
sliderItems.add(new SliderItem(R.drawable.ic_ahorn_blatt));
sliderItems.add(new SliderItem(R.drawable.ic_ahorn_baum));

viewPager2.setAdapter(new SliderAdapter(sliderItems, viewPager2));

viewPager2.setClipToPadding(false);
viewPager2.setClipChildren(false);
viewPager2.setOffscreenPageLimit(3);
viewPager2.getChildAt(index: 0).setOverScrollMode(RecyclerView.OVER_SCROLL_NEVER);
```



Ahorn-Spiele

```
<tree>
  <id>0</id>
  <name>Ahorn</name>
  <profileId>0</profileId>
  <qrCode>Ahorn</qrCode>
  <leafGames>0,300,400</leafGames>
  <fruitGames>70,301,500</fruitGames>
  <trunkGames>40,302,402,501</trunkGames>
  <otherGames>100,303,600,</otherGames>
  <imageTree>ic_ahorn_baum</imageTree>
  <imageLeaf>ic_ahorn_blatt</imageLeaf>
  <imageFruit>ic_ahorn_frucht</imageFruit>
  <imageTrunk>ic_ahorn_stamm</imageTrunk>
  <imageOther>ic_ahorn_stamm</imageOther>
</tree>
```

tees.xml

1. Ahorn

1. Blätter:

1. Quiz:

1. Schau dir die Blätter des Baumes genau an. Welche Form haben sie?
2. Wie sieht der Rand der Blätter aus?
3. Wie sind die Blätter an den Zweigen angeordnet?

2. Foto-Challenge: Nahaufnahme von Blatt.

3. Puzzle: Teile richtig anordnen, sodass sich ein ganzes Ahornblatt ergibt.

4. Blattumriss zeichnen.

2. Rinde:

1. Quiz:

1. Richtige Rinde aus 4 Bildern auswählen.
2. -
3. -

2. Foto-Challenge: Nahaufnahme von Rinde.

3. Puzzle: Im Stamm des abgebildeten Ahorns fehlen z.B. 4 Teile. Die fehlenden Bilder an der richtigen Stelle einfügen (Drag&Drop).

4. Sinne: Rinde erfühlen und beschreiben (Auswahl-Bubble vorhanden).

3. Früchte:

1. Quiz:

1. Wie sehen die Früchte des Ahorns aus? (4 Bilder -> richtiges auswählen)
- 2.
- 3.

2. Fangspiel: Über den Bildschirm fallen verschiedene Früchte. Auf die richtigen klicken und einsammeln.

3. Foto-Challenge: Nasenzwicker auf die Nase kleben und Selfie machen.

4. Reim (Wörter oder Satzteile gegeben, Reim zu Nasenzwicker überlegen)

4. Allgemein:

1. Superkraft des Ahorns.

2. Zuordnungsspiel: In der Mitte der Baum. Außen herum verschiedene Blätter/Blüten/Früchte -> die zum Ahorn dazugehörige Elemente auf Baum ziehen.

3. Sammeln: Blätter/Früchte etc. von Ahorn sammeln und etwas basteln -> Foto (z.B. Blumenstrauß aus bunten Herbstblättern)

4. Memory: Pärchen von Ahorn-Elementen finden (Früchte, Blätter, Rinde, ganzer Baum).

Impressum

```
1 package de.lmu.treeapp.activities;
2
3 import ...
4
5
6
7
8
9
10
11 public class Imprint extends AppCompatActivity {
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15
16         setContentView(R.layout.activity_imprint);
17         super.onCreate(savedInstanceState);
18
19         //Button btn_forward = findViewById(R.id.game_onlyDescription_sendButton);
20
21         TextView imprint_title = findViewById(R.id.imprint_title);
22         TextView imprint_text = findViewById(R.id.imprint_text);
23         TextView publisher_title = findViewById(R.id.publisher_title);
24         TextView publisher_text = findViewById(R.id.publisher_text);
25         TextView authority_title = findViewById(R.id.authority_title);
26         TextView authority_text = findViewById(R.id.authority_text);
27         TextView tax_id_title = findViewById(R.id.tax_id_title);
28         TextView tax_id_text = findViewById(R.id.tax_id_text);
29         TextView tec_implementation_title = findViewById(R.id.tec_implementation_title);
30         TextView tec_implementation_text = findViewById(R.id.tec_implementation_text);
31         TextView text_title = findViewById(R.id.text_title);
32         TextView text_text = findViewById(R.id.text_text);
33         TextView graphics_title = findViewById(R.id.graphics_title);
34         TextView graphics_text = findViewById(R.id.graphics_text);
35         TextView disclaimer_title = findViewById(R.id.disclaimer_title);
36         TextView disclaimer_text = findViewById(R.id.disclaimer_text);
37         TextView copyright_title = findViewById(R.id.copyright_title);
38         TextView copyright_text = findViewById(R.id.copyright_text);
39         TextView privacy_title = findViewById(R.id.privacy_title);
40         TextView privacy_text = findViewById(R.id.privacy_text);
41     }
42 }
```

App ins Grüne

IMPRESSUM

Informationen gemäß § 5 Telemediengesetz und § 55 Rundfunkstaatsvertrag. Die Ludwig-Maximilians-Universität München ist eine Körperschaft des öffentlichen Rechts. Sie wird durch den Präsidenten Prof. Dr. Bernd Huber gesetzlich vertreten.

Herausgeber

Ludwig-Maximilians-Universität München
Geschwister-Scholl-Platz 1 80539 München Telefon:
+49 (0) 89 / 2180 - 0 Kontakt: Poststelle@Verwaltung
.Uni-Muenchen.de Vertretungsberechtigter: Vorname
Nachname

Zuständige Aufsichtsbehörde

Bayerisches Staatsministerium für Wissenschaft und
Kunst Salvatorstraße 2 80327 München Webauftritt:
https://www.km.bayern.de/index_wm.html

Umsatzsteuer-Identifikationsnummer der LMU

Umsatzsteuer-Identifikationsnummer gemäß § 27 a
Umsatzsteuergesetz: DE811205325

Verantwortlich für die technische Umsetzung

Malin Eiband, Yannick Weiss, Lisa Görtz, Cara Emberger

Texte

Thomas Gerl, Marie, Lisa Görtz

App ins Grüne

Grafiken

Cara Emberger, Maya Czipor, Fabienne Lengweiler

Freistellungserklärung

Die Angaben wurden nach bestem Wissen erstellt, Fehler können jedoch nicht mit letzter Sicherheit ausgeschlossen werden. Rechtlich verbindlich sind ausschließlich die Festlegungen in den einschlägigen Rechtsgrundlagen (Gesetze, Verordnungen, Satzungen). Eine Gewähr für die jederzeitige Aktualität, Richtigkeit, Vollständigkeit und Verfügbarkeit der bereitgestellten Informationen können wir nicht übernehmen. Wir haften nicht für Schäden, die durch die Nutzung dieses Appangebots entstehen. Dieser Haftungsausschluss gilt nicht, soweit die Vorschriften des § 839 BGB (Haftung bei Amtspflichtverletzung) einschlägig sind. Für etwaige Schäden, die beim Aufrufen oder Herunterladen von Daten durch Schadsoftware oder der Installation oder Nutzung von Software verursacht werden, wird nicht gehaftet. Der Haftungsausschluss gilt nicht für Informationen, die in den Anwendungsbereich der Verordnung (EU) 2016/679 des Europäischen Parlaments und des Rates vom 27. April 2016 (Datenschutz-Grundverordnung) fallen. Für diese Informationen wird die Richtigkeit und Aktualität gewährleistet.

Copyright / Urheberrecht

© Ludwig-Maximilians-Universität München - alle Rechte vorbehalten. Texte, Bilder, Grafiken sowie die Gestaltung

App ins Grüne

bereitgestellten Informationen können wir nicht übernehmen. Wir haften nicht für Schäden, die durch die Nutzung dieses Appangebots entstehen. Dieser Haftungsausschluss gilt nicht, soweit die Vorschriften des § 839 BGB (Haftung bei Amtspflichtverletzung) einschlägig sind. Für etwaige Schäden, die beim Aufrufen oder Herunterladen von Daten durch Schadsoftware oder der Installation oder Nutzung von Software verursacht werden, wird nicht gehaftet. Der Haftungsausschluss gilt nicht für Informationen, die in den Anwendungsbereich der Verordnung (EU) 2016/679 des Europäischen Parlaments und des Rates vom 27. April 2016 (Datenschutz-Grundverordnung) fallen. Für diese Informationen wird die Richtigkeit und Aktualität gewährleistet.

Copyright / Urheberrecht

© Ludwig-Maximilians-Universität München - alle Rechte vorbehalten. Texte, Bilder, Grafiken sowie die Gestaltung unterliegen dem Urheberrecht. Sie dürfen nur zum privaten und sonstigen eigenen Gebrauch im Rahmen des §53 Urheberrechtsgesetz (UrhG) verwendet werden. Eine anderweitige Verwendung ist nur nach vorheriger Genehmigung gestattet.

Datenschutzerklärung

Rechtsgrundlagen der Datenverarbeitung, Empfänger der personenbezogenen Daten, Rechte der betroffenen Person, Widerrufsrecht bei Einwilligung, Erhebung von Daten

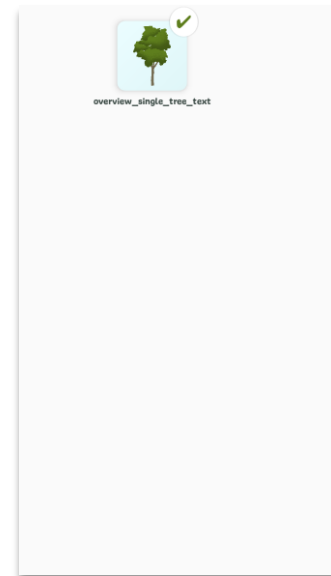
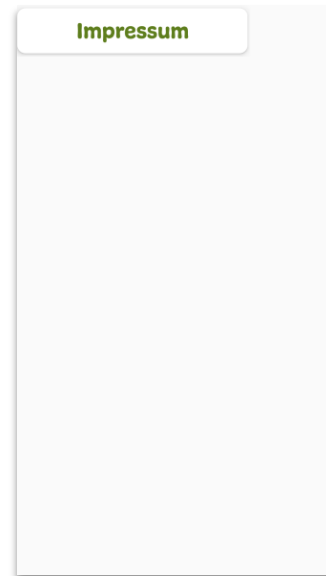
Beschreibungen

Strings.xml

```
39 <string name="game_take_picture_example">Beispielbild</string>
40 <string name="game_take_picture_own_foto">Vorschau deines Fotos</string>
41 <string name="game_take_picture_send_button_placeholder_text">Senden</string>
42 <string name="game_selection_headline_placeholder">Wähle ein Spiel aus</string>
43 <string name="game_selection_description_placeholder">Hier findest du Minigames zu Rinde und Holz der Waldkiefer. Starte jetzt mit einem Spiel
44 <string name="detail_single_tree_profile_button_text">Steckbrief</string>
45 <string name="detail_single_tree_leaf_button_content_description">leaf games button</string>
46 <string name="detail_single_tree_fruit_button_content_description">fruit games button</string>
47 <string name="detail_single_tree_trunk_button_content_description">trunk games button</string>
48 <string name="detail_single_tree_other_button_content_description">other games button</string>
49 <string name="popup_quiz_positive_title">Richtig!</string>
50 <string name="popup_quiz_positive_text">Dein Foto erscheint jetzt auch im Steckbrief!</string>
51 <string name="popup_quiz_negative_title">Leider falsch!</string>
52 <string name="popup_quiz_negative_text">Die richtige Antwort ist:</string>
53 <string name="popup_neutral_btn_wiki">Steckbrief</string>
54 <string name="popup_neutral_btn_forward">Fertig!</string>
55 <string name="popup_neutral_title">Prima!</string>
56 <string name="imprint_headline1_placeholder">IMPRESSUM</string>
57 <string name="imprint_description1_placeholder">Informationen gemäß § 5 Telemediengesetz und § 55 Rundfunkstaatsvertrag. Die Ludwig-Maximilian-
58 <string name="imprint_headline2_placeholder">Herausgeber</string>
59 <string name="imprint_publisher_description">
60     Ludwig-Maximilians-Universität München&#10;
61     Geschwister-Scholl-Platz 1&#10;
62     80539 München&#10;
63     Telefon: +49 (0) 89 / 2180 - 0&#10;
64     Kontakt: Poststelle@Verwaltung.Uni-Muenchen.de&#10;
65     Vertretungsberechtigter: Vorname Nachname</string>
66 </resources>
```

Impressum

| Deine Waldübersicht | | |
|---|--------|--------|
| Klicke auf einen Baum um mehr über ihn zu erfahren! | | |
| Item 0 | Item 1 | Item 2 |
| Item 3 | Item 4 | Item 5 |
| Item 6 | Item 7 | Item 8 |
| Item 9 | | |



overview_button_imprint.xml

```
// Replace the contents of a view (invoked by the layout manager)
@Override
public void onBindViewHolder(ViewHolder holder, final int position) {

    if(position == treeValues.size()) {

        holder.btn_imprint.setOnClickListener((view) -> {
            Intent intent_imprint = new Intent(context, Imprint.class);
            context.startActivity(intent_imprint);
        });

    } else {
```

OverviewRecyclerViewAdapter.java



Splash Screen

```
activity_imprint.xml × activity_splash_screen.xml × strings.xml × Imprint.java × SplashScreen.java ×
1 package de.lmu.treeapp.activities;
2
3 import ...
13
14 public class SplashScreen extends AppCompatActivity {
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         ActionBar actionBar = getSupportActionBar();
20         actionBar.hide();
21         setContentView(R.layout.activity_splash_screen);
22
23
24         ImageView image_appinsgruene = findViewById(R.id.image_appinsgruene);
25         ImageView image_logos = findViewById(R.id.image_logos);
26         TextView splash_screen_text = findViewById(R.id.splash_screen_text);
27
28         new Handler().postDelayed(() -> {
29             Intent intent = new Intent( packageName: SplashScreen.this, MainActivity.class);
30             startActivity(intent);
31             finish();
32         }, delayMillis: 3000);
33
34     }
35 }
36
37
38
```



Äußere Anforderungen

- Emulator Probleme
- Vorschau nicht gleich Emulator
- GitHub super!

Wie geht's weiter?

Empfehlenwert: Fortführung im größeren Team

- + Aufgabenverteilung
- + effektive Umsetzung
- + Fertigstellung in kürzerer Zeit



Möglichkeit:

- Folgeseminar der LMU (evtl. mit Projektleitung unsererseits)
- Non-profit Organization für digitale Bildung
- Start-Up