

Esplora, Impara, con  
TEDx e le lingue!

# LinguaLearnX

**Tomasoni Francesco - 1080980**

Homework 4

# Home Screen

Avviata l'applicazione, è possibile inserire il topic di interesse nella barra di ricerca e scegliere la lingua desiderata.

Premendo il tasto con la lente di ingrandimento, verranno mostrati i titoli dei talk relativi al topic scelto.

The screenshots illustrate the LinguaLearnX app's user interface across three stages:

- Search by tag:** The first screen shows a dropdown menu titled "Select Language" listing ten languages: English, German, French, Spanish, Italian, Portuguese, Dutch, Russian, Chinese, and Japanese. Above the dropdown is a search bar with the placeholder "Search by tag".
- Filtered results:** The second screen shows the same search bar with the tag "design" entered. Below it, a list of video titles is displayed, all filtered by the selected language (English). The visible titles are: "How do gas masks actually work?", "Ideas change everything – and what's next for TED", "A futuristic vision for Latin America, rooted in ancient design", "How do gas masks actually work?", "Ideas change everything – and what's next for TED", "A futuristic vision for Latin America, rooted in ancient design", "What's the point of digital fashion?", and "TED is 40 – here's how it all started".
- Video details:** The third screen shows the full details for a specific video titled "How do gas masks actually work?". The page includes a "Description" section with a detailed paragraph about gas masks and their effectiveness, and a "Recommended Videos" section featuring other TED talks like "The small business hustle" and "Relief to recovery".

```
Future<void> fetchVideos(String tag) async {
final response = await http.post(
  Uri.parse('https://ednortd1sa.execute-api.us-east-1.amazonaws.com/default/Get_Talks_By_Tag'),
  headers: {'Content-Type': 'application/json'},
  body: json.encode({'tag': tag}),
);

if (response.statusCode == 200) {
  final data = json.decode(utf8.decode(response.bodyBytes)) as List;
  setState(() {
    _videos = data.map((video) => Video.fromJson(video)).toList();
  });
} else {
  throw Exception('Failed to load videos');
}
}
```

# Lista dei video cercati tramite tag

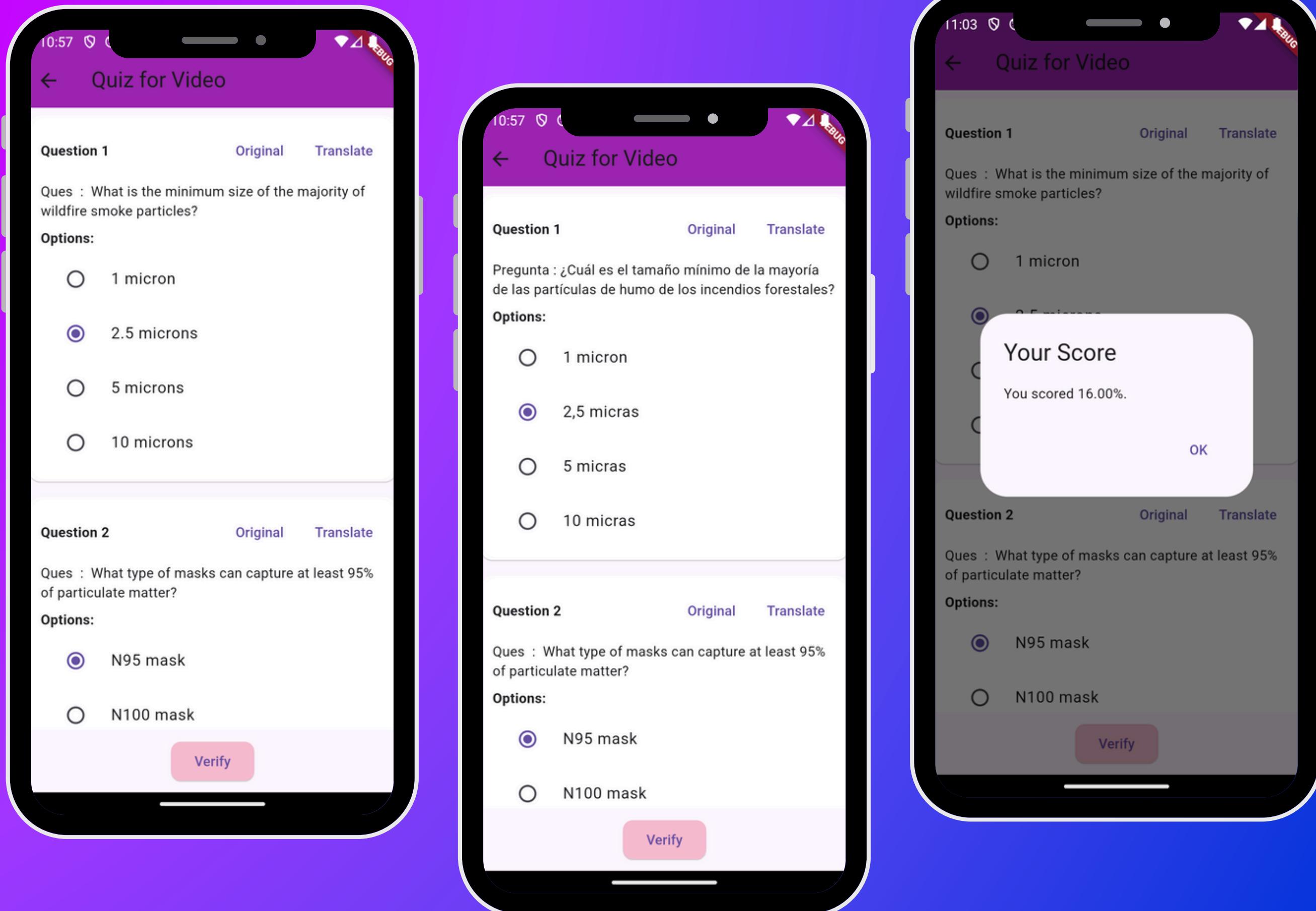
# Lista dei video whatch next

```
Future<List<Video>> fetchRelatedVideos(String videoId) async {
try {
  final response = await http.post(
    Uri.parse('https://lzhjxdkdfe.execute-api.us-east-1.amazonaws.com/default/Get_Watch_Next_by_Idx'),
    headers: {'Content-Type': 'application/json'},
    body: json.encode({'id': videoId}),
  );

  if (response.statusCode == 200) {
    final data = json.decode(utf8.decode(response.bodyBytes)) as List;
    List<Video> videos = data.map((item) => Video.fromJson(item)).toList();
    return videos;
  } else {
    throw Exception('Failed to load related videos: ${response.statusCode}');
  }
} catch (e) {
  throw Exception('Failed to load related videos: $e');
}
}
```

# Quiz page

Nella sezione quiz, sono presenti domande con le relative opzioni, che se cliccato il pulsante "translate", vengono tradotte nella lingua scelta inizialmente tramite API di Deepl.  
Il pulsante verifica restituirà il punteggio in percentuale



```
Future<List<dynamic>> fetchQuiz() async {
final response = await http.post(
  Uri.parse('https://ak324jkhie.execute-api.us-east-1.amazonaws.com/default/Get_Quiz_By_Id_function'),
  headers: {'Content-Type': 'application/json'},
  body: json.encode({'id': widget.video.idRef}),
);

if (response.statusCode == 200) {
  final data = json.decode(utf8.decode(response.bodyBytes));
  if (data.isEmpty) {
    return await generateQuiz();
  } else {
    return data;
  }
} else {
  throw Exception('Failed to load quiz');
}
}
```

## caricamento dei quiz presenti nel database

## generazione dei quiz non ancora generati

```
Future<List<dynamic>> generateQuiz() async {
final response = await http.post(
  Uri.parse('https://v0mtx7aipf.execute-api.us-east-1.amazonaws.com/default/Get_Quiz_By_Id'),
  headers: {'Content-Type': 'application/json'},
  body: json.encode({'id': widget.video.idRef}),
);

if (response.statusCode == 200) {
  final data = json.decode(utf8.decode(response.bodyBytes));
  return data;
} else {
  throw Exception('Failed to generate quiz');
}
}
```

# DeepL API



L'API gratuita consente di tradurre testo in qualsiasi lingua supportata, ma presenta alcune limitazioni, come un numero massimo di caratteri mensili.

```
Future<List<String>> translateOptions(List<String> options) async {
    final response = await http.post(
        Uri.parse('https://api-free.deepl.com/v2/translate'),
        headers: {
            'Authorization': 'Type: String ey 2b49b42d-74b5-448e-b67c-09e3a0650153:fx',
            'Content-Type': 'application/json',
        },
        body: json.encode({
            'text': options,
            'target_lang': widget.language,
        }),
    );

    if (response.statusCode == 200) {
        final data = json.decode(utf8.decode(response.bodyBytes));
        return List<String>.from(data['translations'].map((translation) => translation['text']));
    } else {
        print('Failed to translate options. Status code: ${response.statusCode}');
        print('Response body: ${response.body}');
        throw Exception('Failed to translate options');
    }
}
```

# Criticità tecniche:

## Traduzioni in tempo reale:

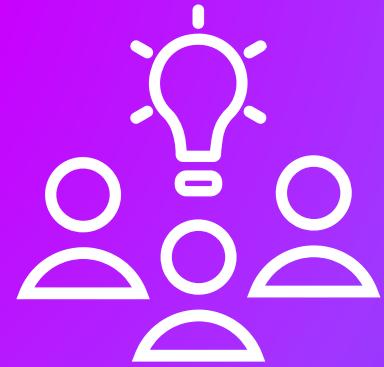
Il progetto dipende dall'API di traduzione di terze parti (DeepL) per tradurre testi in tempo reale, introducendo rischi come ritardi, costi, e possibili interruzioni del servizio.

## Performance e scalabilità:

La gestione delle richieste HTTP per caricare i quiz da un server remoto può causare problemi di performance e scalabilità, specialmente con un aumento significativo degli utenti o con connessioni Internet lente.

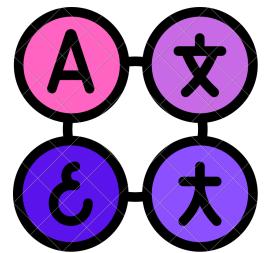
## Gestione dello stato e delle risposte:

La gestione dello stato locale per le risposte degli utenti può diventare complessa con un numero crescente di domande o funzionalità aggiuntive, rendendo difficile il recupero dei progressi in caso di interruzioni o chiusura dell'app.



## Possibili implementazioni future

- 01 Visualizzazione effettiva dei video TedX
- 02 Miglioramento della UX/UI, come da mockup creato inizialmente
- 03 Creazione di una sezione “continua a guardare” e “sfida gli amici”



Esplora, Impara, con  
TEDx e le lingue!

# LinguaLearnX

