Import Reminders from Reminders App

Problem

Todoist app users who previously used Apple's Reminders app currently don't have an opportunity to seamlessly transfer their reminders/to-do tasks from Apple's Reminders app to Todoist app.
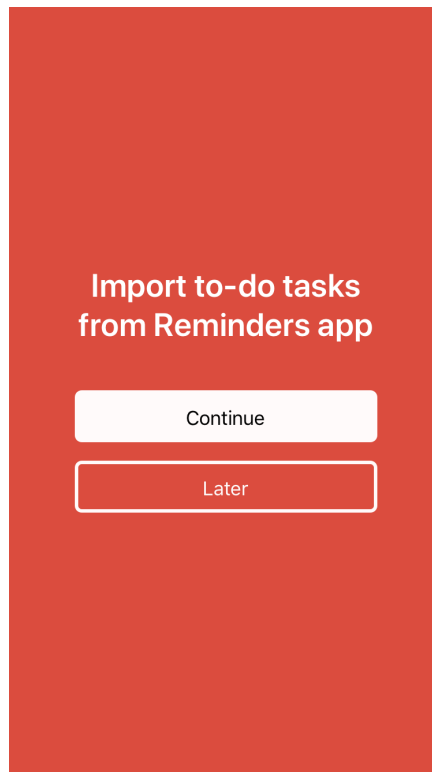
The only way to export reminders from Reminders app is limited to MacOS version of this app (*.ics format) — there is no chance the user can export reminders in iOS. Regardless of that, Todoist app doesn't allow to import reminders in *.ics format.
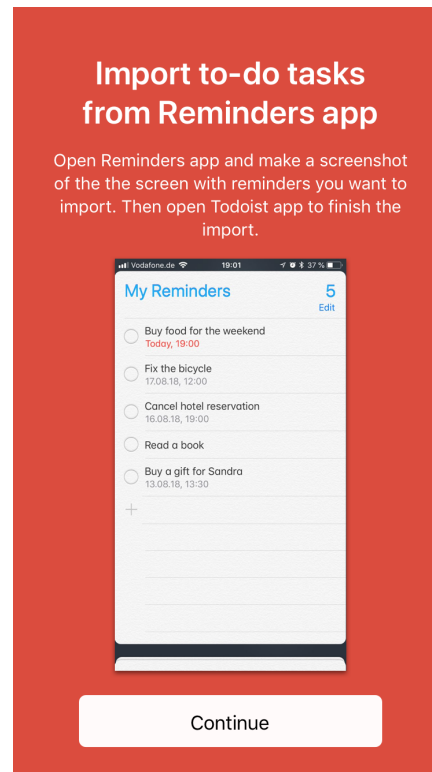
Solution

Todoist app can implement reminders/to-do tasks export workflow as a part of the onboarding workflow.

The export workflow is based on the *text recognition from image* approach. There are many available solutions leveraging machine learning techniques like Amazon Rekognition, Microsoft Computer Vision, Google Optical Character Recognition and other.
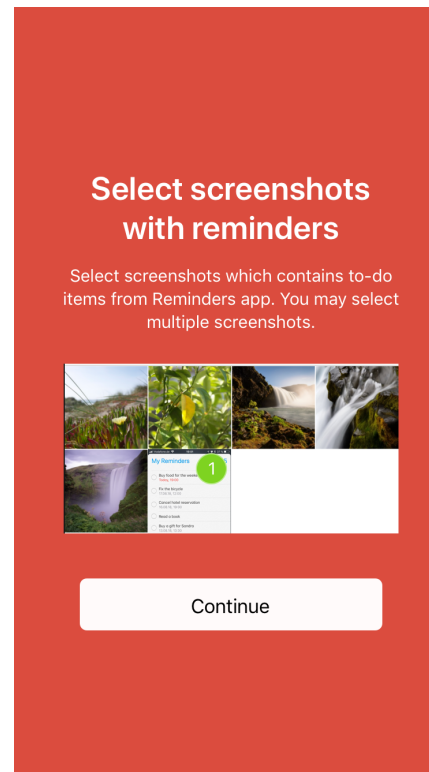
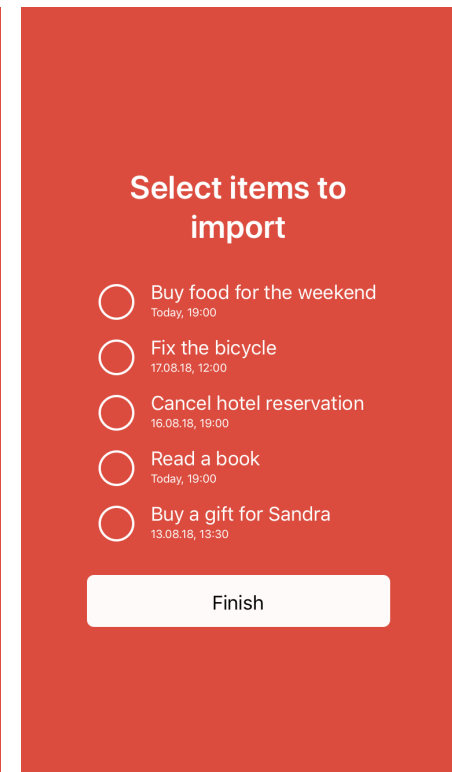*Screen #1, user is requested to import tasks during onboarding*

Screen #2, user is instructed on how to proceed further

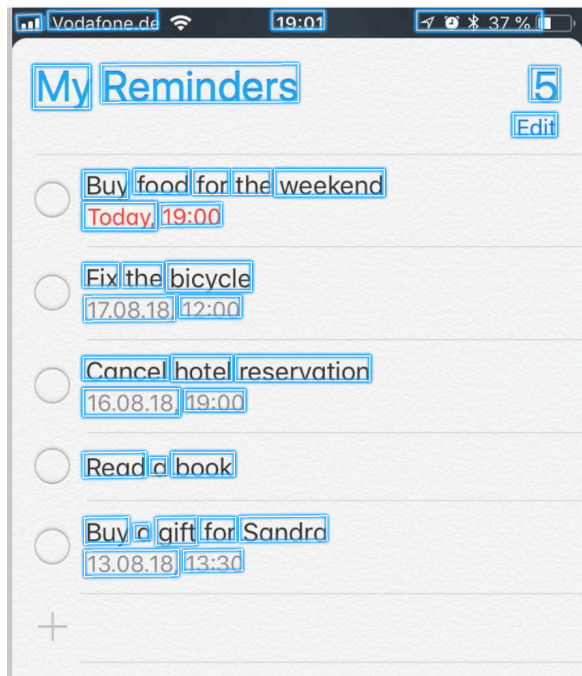*Screen #3, user selects screenshots from photo gallery*

Screen #4, user selects items to import

On screen #3 user uploads screenshots containing to-do items to Todoist backend which calls Amazon Recognition API's *DetectText Operation* on the uploaded screenshot. After processing the image, Amazon Recognition API responds with text recognized on the screenshot. The response provides additional information allowing to identify text object position on the screenshot relatively to other text objects present on the same screenshot. This enables Todoist backend to parse the response and extract to-do item. As the last step, Todoist backend responds to Todoist app with a collection of to-do items retrieved earlier.

Text elements on screenshot as seen by Amazon Recognition API	Visual representation of recognized text objects	Part of response generated by Amazon Recognition API


```
| ool | Vodafone.de | 19:01 | 4037%C |        ...
                                              {
   | My | Reminders | 5 |
                                                  "Confidence": 99.753662109375,
        | Edit |
                                                  "DetectedText": "Buy food for the weekend",
| Buy | food | for | the | weekend |
                                                  "Geometry": {
      | Today, | 19:00 |
                                                      "BoundingBox": {
     | Fix | the | bicycle |
                                                          "Height": 0.029414251446723938,
    | 17.08.18, | 12:00 |
                                                          "Left": 0.12210245430469513,
| Cancel | hotel | reservation |
                                                          "Top": 0.1588670015335083,
     | 16.08.18, | 19:00 |
                                                          "Width": 0.5287880897521973
     | Read | a | book |
                                                      },
| Buy | a | gift | for | Sandra |                      ...
     | 13.08.18, | 13:30 |                        },
                                                  "Id": 3,
                                                  "Type": "LINE"
                                              },
                                                  ...
```

With minimal adaptations, this approach could be applied to screenshots with other screen orientations, background/text colors, etc.