

GitHub Username: AhmedNader65

Pro autocue - Android teleprompter

Description

Type or import text and easily teleprompt it, Read speeches or scripts with confidence
As It allows a presenter/content creator to read a script whilst maintaining direct eye contact with the audience/camera. Because the speaker does not need to look down to consult written notes, he/she appears to have memorized the speech or to be speaking spontaneously.

Intended User

youtubers, filmmakers, videomakers, musicians & singers, business professionals, presentations, lectures, broadcasters or any public speaking

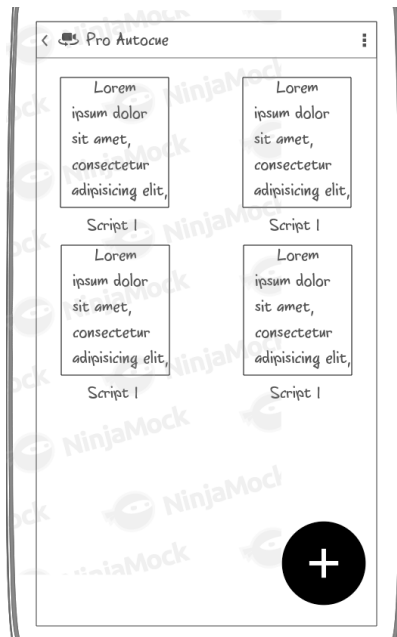
Features

- Import text scripts
- Edit, delete and save scripts
- Control scrolling speed
- Change text color, font size, background color, Line height
- Use mobile camera to record videos with on screen teleprompter
- Control in-app screen brightness
- Remote control (Via bluetooth) :- (maybe :D)
- Mirror mode
- Disable screen timeout
- Use gestures to control speed and size (maybe :D)

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1



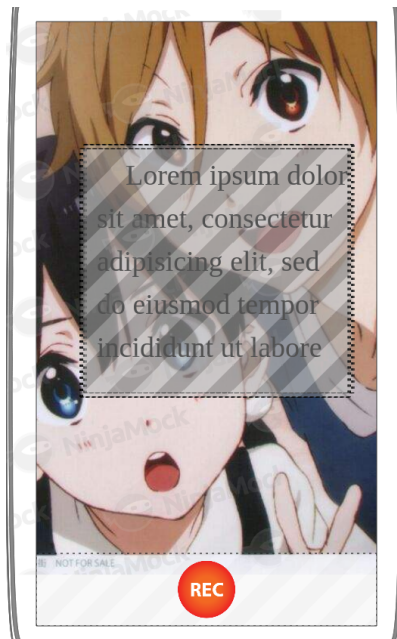
This screen to select an already used before script or import a new script

Screen 2



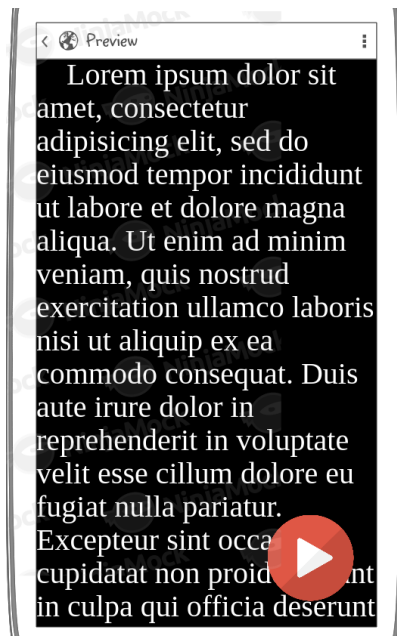
To determine whether the user wants to use the mobile to record or just as a teleprompter

Screen 3



When the user select to use the recorder it will display the script on the screen above the camera view

Screen 4



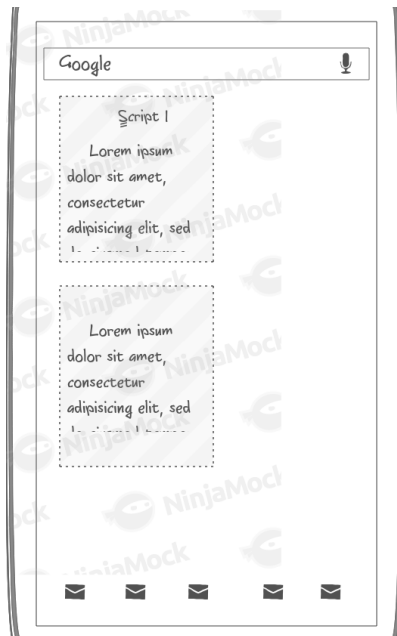
This just a preview and for play the script if the user doesn't want to use the app recorder

Screen 5



The settings screen to customize how the app works

Screen 6



Widget (user can choose a script to be on the main screen and quickly access it)

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

I will build a Content Provider to handle the previously used scripts

Describe any edge or corner cases in the UX.

The user can use previously used scripts directly from the app
The user can change font size and scrolling speed by gestures
User can change the view on the camera (size and place)

Describe any libraries you'll be using and share your reasoning for including them.

Android support library :- recycler-view , cardView, coordinateLayout, FAB

Android-gesture-detectors :- [github](#) for gesture recognize

AdMob :- to provide ads for free version [FIREBASE AdMob](#)

Google Analytics via [FIREBASE](#)

Describe how you will implement Google Play Services or other external services.

Will use Google Mobile Ads to display ads on the free version of the app

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Create the app module, configure min sdk

- Add and configure the libraries

If it helps, imagine you are describing these tasks to a friend who wants to follow along and build this app with you.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for ScriptsSelectionActivity,RecordOrTeleActivity,PreviewActivity and RecordActivity

Task 3: implement ScriptsSelectionActivity

- Implement content provider and use Loaders to load the data into the recyclerview
- Implement file selection

Task 4: Camera

- Implement the using of Camera in the app and put the textView on the Frame while recording

Task 5: Settings

- Implement the Settings activity and how it affects the Teleprompter preview and change how the app works

Task 6: use Gestures

- Implement the gestures and use it to control the speed and size of the text

Task 7: use FirebaseJobDispatcher

- User JobDispatcher to send periodic notifications

Task 8: implement google Analytics

Task 9 : build variants

- Build 2 variants (Free : with ads and Paid without ads)