Rezolve Gist Commenter

Design, Environment setup and code execution instructions

09/19/2018

Table of Contents

1 Introduction

This document provides detailed information about the assignment along with all the instruction and workflows covered in the application.

2 Assignment Details

This test exercises different aspects of your technical abilities when faced with requirements and expectations. The delivery shall be an intuitive and user-friendly interface capable of keeping the user's attention by its visual appearance and usability. It should also be a concise, clear and organized code overall.

2.1 Definition

1. Develop an app that allows users to comment on a Github Gist.
2. The user must open Gist through a QRCode scan.
3. Only after Gist opens the user should be able to comment on it.

2.2 Expected Output

1. Develop a prototype to solve the problem above
2. Use of the Github GIST API is mandatory. This can be accessed through the following link: https://developer.github.com/v3/gists/

3 Application Workflows

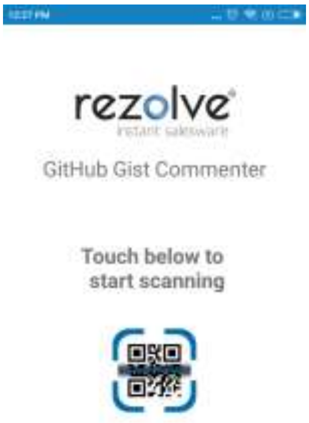
Based on the requirement in section 2 above, the application is designed and developed to cater to the problem statement. Below are the detailed feature and functional flows implemented.

3.1 UI Flow

1. On launch of application the "Home Screen" will be displayed to the user.
2. If the user does not click on the scan code icon within 3 seconds an audio voice will be played to assist the user to perform action.
3. On clicking the scan icon user will be redirected to the "Scan QR Screen".
4. User can scan a valid QR code and upon successful QR scan user will be redirected the respective "Gist Info Screen".
5. User can view the gist information along with all the comments on this gist.
6. User can comment on this gist using the "Leave a comment" option. The user needs to provide his GitHub login credentials as this is mandatory for commenting on any gist.
7. On successful authentication the comment will be posted and displayed on the gist.

3.2 App Screens and Description

3.2.1 Home Screen

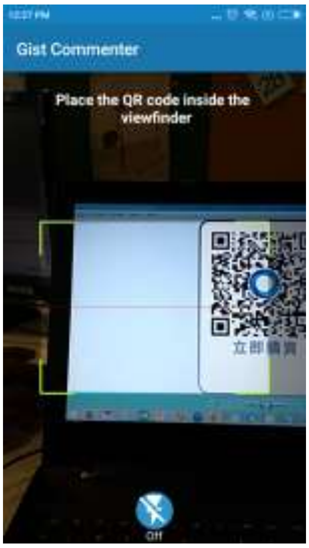


GitHub Gist Commenter

Touch below to start scanning

1. This is the application launch screen as displayed above.
2. This screen displays application logo and app name followed by a scan icon to scan the QR code.
3. In case the user does not click on “Scan icon” until 3 second, a voice based assistance will be played to help the user take action in this screen.
4. The Touch icon has been implements with an animation every 3 seconds.

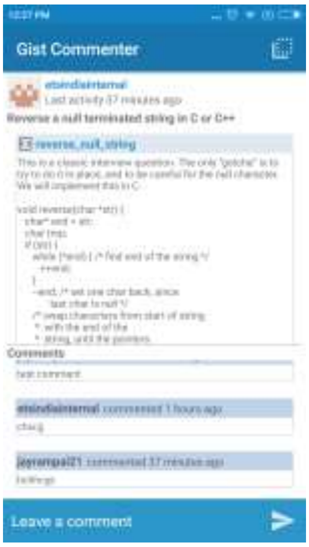
3.2.2 Scan QR Screen



Place the QR code inside the viewfinder

1. This screen is launched when user clicks on the “Scan icon” on Home page or rescan icon on “Gist Info Screen”
2. User can scan a QR code on this screen.
3. On successful syntax match (Refer section 3 below) the user will be redirected to the Gist Info screen with gist information displayed
4. On syntax mismatch the user will have option to read the QR data and proceed accordingly.
5. In case of bad light conditions the user can use the flash feature as needed.

3.2.3 Gist Info Screen



Gist Commenter

etsindiainternal
Last activity 37 minutes ago

Reverse a null terminated string in C or C++

reverse_null_string

This is a classic interview question. The only "gotcha" is to try to do it in place, and to be careful for the null character. We will implement this in C.

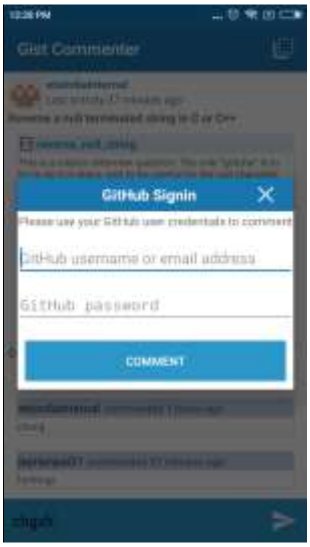
```
void reverse(char *str) {
  char* end = str;
  char tmp;
  if (str) {
    while (*end) { /* find end of the string */
      ++end;
    }
    --end; /* set one char back, since
        last char is null */
    /* swap characters from start of string
     * with the end of the
     * string, until the pointers
```

Comments

test comment

etsindiainternal commented 1 hours ago

jayrampal21 commented 37 minutes ago



Gist Commenter

GitHub Signin

Please use your GitHub user credentials to comment

1. This screen displays all the gist information along with comments.
2. Screen is divided into 2 scrollable sections:
    a. Gist Info: This displays the information about the gist which include
        i. GitHub username of owner/creator of gist.
        ii. Last activity on gist.
        iii. Description/Title of this gist.
        iv. Profile picture of gist owner.
        v. Display of each file under this gist with respective file names
    b. Comments: This section displays all the comments on the gist which include
        i. GitHub username of commented user
        ii. Comment timeframe
        iii. Comment body
3. This screen also has a “Leave a comment” option using which a new comment can be posted on the gist.
4. User needs to provide valid GitHub credentials to post a comment on the gist.
5. This screen has a “Rescan” icon on top right corner.

4 QR Code generation syntax:

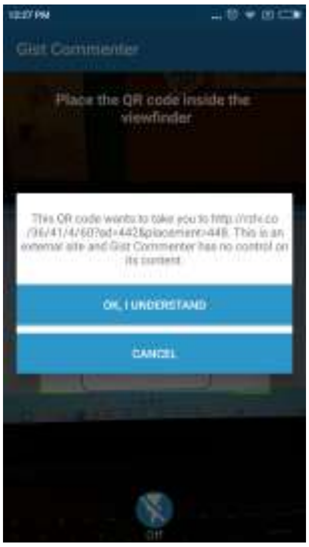
User defined syntax has been used to identify valid QR codes which this application will accept as valid.

Syntax for QR generation: **gid-gist:gist_id**

gid-gist: – every valid QR must start with this keyword.
gist_id – these is the gist id which we can get from gist

e.g. 1: On valid scan, data will read as **gid-gist:4f2c3a4179e95fa425aa2d34264dafe2**

e.g. 2: On invalid scan the user will the below screen



12:27 PM

Gist Commenter

Place the QR code inside the viewfinder

This QR code wants to take you to http://rdfx.co/96/41/4/60?ad=442&placement=448. This is an external site and Gist Commenter has no control on its content.

5 Code Explained

Techniques and libraries as listed below have been used to implement this functionality.

1. Developed using Android Studio 3.1 and Java 8
2. Basic MVC coding architecture has been used
3. Below android components have been used
    i. Activities, ListView, CardView and other android UI components.
    ii. AsyncTask, MediaPlayer, Adapters, Callback Listeners, Android UI thread
    iii. Used Butterknife for view data binding.
    iv. Singleton Patterns for constants
4. Libraries:
    a. ZXingScanner library for QR code scanning.
    b. OkHTTP3 for HTTP request handling
    c. Picasso2.5 library for image rendering.
5. GitHub API: https://developer.github.com/v3/gists/ API has been used as below
   All the below APIs use the basic authentication method for communication
    a. Get Gist: To get all the gist specific information like owner, files etc
       API: https://api.github.com/gists/:gistid
       Method: GET
       Authentication: Basic Auth
    b. Get Gist comments: To get all the gist specific comments
       API: https://api.github.com/gists/gistid/comments
       Method: GET
       Authentication: Basic Auth
    c. Create a gist comment: Create a new comment on gist using valid GitHub credentials
       API: https://api.github.com/gists/gistid/comments
       Method: POST
       Authentication: Basic Auth
       Body:

```
{
  "body": "Comment Description"
}
```

6. UI Design: We have used combination of google material design (Card layout) along with some default android UI components.

6 Steps to Compile Code

1. Prerequisites: List of all tools and software which are mandatory for the code to run.
    a. Android Studio 3.1 needs to be installed.
    b. Java 8 needs to be installed.
2. Download the code from this GitHub repo https://github.com/etsindiainternal/gistcommenter
3. Launch the Android studio 3.1 and select Import Projects
4. Import the project codebase from **QRCodeScanner folder** available in point 2.
5. Run the project using the “Run” command

7 Deliverables

Below deliverable are provided along with this document.

1. Code for this assignment can be accessed using the URL from GitHub
   https://github.com/etsindiainternal/gistcommenter
   Folder named **QRCodeScanner** consist complete android app codebase
2. Application APK file can be installed using the file located in **APK** folder
3. Sample valid QR codes, this can be located in **QR Codes** folders
4. Sample invalid QR codes, this can be located in **QR Codes** folders

8 Compatibility Info

Only normal and large screen layout has been implemented. App has been tested on small, normal and large device screens. The application works smoothly across all these screen sizes however due to time constraint other layout designs have not been implemented (design layout for small, xlarge as per the style guidelines can be implemented if required).

OS version supported: version 5.0 and above are supported.

9 Enhancements that we suggest

1. Login feature to this app so that the user does not have to provide credentials for every comment. We can use SQLite Cipher to store the user credentials safely.
2. Select QR from gallery feature on the SR scan screen can be implemented using which the user can select a saved QR image.
3. Edit/delete feature for a comment on gist can also be implemented.