 <p>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</p>	<p>Computação Móvel e Ubíqua</p> <p>1º Semestre</p> <p>Docente: fas</p> <p>Ficha Prática: Firebase Libraries</p>
---	---

Tema: *Firebase Libraries*

Objetivos:

- Utilização de *Firebase Libraries*
- Autenticação
- Partilha de dados entre utilizadores

Exercício 1


Utilizando o IDE Android Studio crie um novo projeto com as seguintes características:

<i>Atributo</i>	<i>Valor</i>
<i>Application Name</i>	AuthenticationApplication
Project Location	Leave the default value
Form factor	Phone and tablet only
Minimum SDK	API 23 Marshmallow
Type of Activity	Empty
Activity Name	MainActivity (default)
Layout Name	Activity_main (default)

Neste exercício pretende-se o desenvolvimento de um menu de login utilizando aplicação que permita a implementação de um menu de autenticação de utilizadores baseando-se em firebase authentication. Deverá usar os recursos da plataforma Firebase¹ para autenticar utilizadores já registados ou criar um novo registo de utilizador.

Após a autenticação a aplicação deverá redirecionar o utilizador para a parte privada da aplicação voltando ao menu de login assim que o utilizador execute a ação logout.

¹ Documentação sobre Autenticação na plataforma Firebase: <https://firebase.google.com/docs/auth/android/password-auth>

 <p>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</p>	<p>Computação Móvel e Ubíqua</p> <p>1º Semestre</p> <p>Docente: fas</p> <p>Ficha Prática: Firebase Libraries</p>
---	---

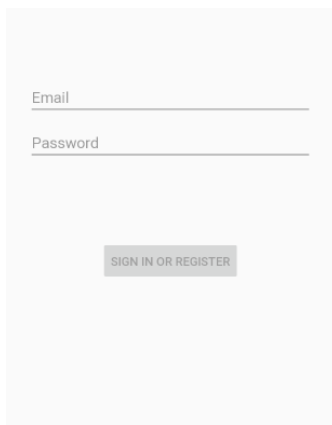


Figura 1 - Exemplo do menu de login

Exercício 2

Utilizando a aplicação do exercício 2, após o login efetuado devemos criar uma atividade que permita guardar imagens capturadas pela aplicação definida para fotos no dispositivo android. Para isso deve usar o mecanismo *startActivityForResult* que passa um *intent* ao sistema para que este inicie a ação pretendida numa aplicação externa que deve retornar o resultado à nossa aplicação. Para ajuda atente ao excerto de código disponibilizado na tabela 1 para iniciar a captura de uma imagem e obter o resultado.

Tabela 1- Excerto para obter uma imagem a partir da aplicação predefinida em Android

```
static final int REQUEST_IMAGE_CAPTURE = 1;

private void dispatchTakePictureIntent() {
    Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
        startActivityForResult(takePictureIntent, REQUEST_IMAGE_CAPTURE);
    }
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == REQUEST_IMAGE_CAPTURE && resultCode == RESULT_OK) {
        Bundle extras = data.getExtras();
        Bitmap imageBitmap = (Bitmap) extras.get("data");
        imageView.setImageBitmap(imageBitmap);
    }
}
```

Após obter a imagem resultado, deverá guardar a imagem usando a *firebase storage*,² guardando a referência/nome da imagem numa coleção no *firebase firestore*.³ Por fim, deverá ser possível listar todas as referências/nome das imagens presentes no *firebase firestore* numa lista. Após a seleção do item da lista deve poder ser consultada a imagem respetiva.

Exercício 3

Utilizando o IDE Android Studio crie um novo projeto com as seguintes características:


<i>Atributo</i>	<i>Valor</i>
<i>Application Name</i>	<i>BarCodeScanner</i>
Project Location	Leave the default value
Form factor	Phone and tablet only
Minimum SDK	API 23 Marshmallow
Type of Activity	Empty
Activity Name	MainActivity (default)
Layout Name	Activity_main (default)

Neste exercício pretende-se o desenvolvimento de uma aplicação que permita a utilização do Firebase cujo objetivo é a leitura automática de códigos de barras ou QRcodes. Para este projeto deve realizar os seguintes passos:

1. Criação do projeto Android no computador
2. Adição do firebase
3. Após obtenção da imagem seguir as indicações do código exemplo ML Barcode Scanner disponibilizadas pela Google em <https://firebase.google.com/docs/ml-it/android/readbarcodes?authuser=0>

² Documentação sobre *firebase storage* : <https://firebase.google.com/docs/storage/android/upload-files>

³ Documentação sobre *firebase firestore* : <https://firebase.google.com/docs/firestore/quickstart>

 <p>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</p>	<p>Computação Móvel e Ubíqua</p> <p>1º Semestre</p> <p>Docente: fas</p> <p>Ficha Prática: Firebase Libraries</p>
---	---

4. No momento do reconhecimento do código deverá imprimi-lo através de um Toast ou registar o registo numa coleção do *firebase firestore*

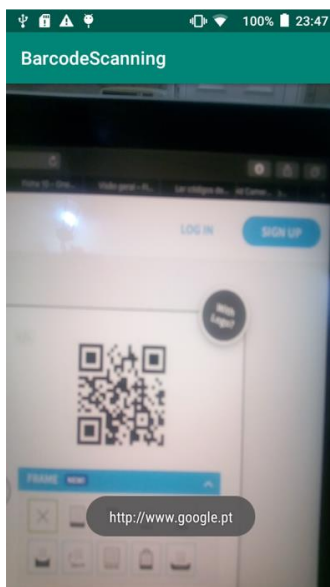


Figura 2 - Exemplo de leitura de código QR

Na aplicação exemplo, a imagem é obtida a partir de uma imagem já presente no dispositivo. Tal não será o efeito desejado numa aplicação quer obter a leitura de códigos de barras em tempo real. Para o efeito iremos adicionar a biblioteca CameraX para obter imagens diretamente a partir da câmara do dispositivo. Para tal é necessário usar uma *TextureView* no layout onde será apresentada a imagem da camara do dispositivo em tempo real e que será analisada a frame a frame. Pretende-se que no final seja possível a captura em tempo real pela aplicação.

Tabela 2 - Excerto para inicialização de pré-visualização e captura de imagens numa TextureView

```
private void startCamera() {
    PreviewConfig pConfig = new PreviewConfig.Builder()
        .setTargetRotation(getWindowManager().getDefaultDisplay().getRotation())
        .build();
    Preview preview = new Preview(pConfig);

    preview.setOnPreviewOutputUpdateListener(
        new Preview.OnPreviewOutputUpdateListener() {
            @Override
            public void onUpdated(Preview.PreviewOutput output) {
                ViewGroup parent = (ViewGroup) textureView.getParent();
                parent.removeView(textureView);
                parent.addView(textureView, 0);
                textureView.setSurfaceTexture(output.getSurfaceTexture());
            }
        }
    );
}
```

 <p>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</p>	<p>Computação Móvel e Ubíqua</p> <p>1º Semestre</p> <p>Docente: fas</p> <p>Ficha Prática: Firebase Libraries</p>
---	---

```

    }
    });

```

```

    ImageCaptureConfig imageCaptureConfig = new
    ImageCaptureConfig.Builder().setCaptureMode(ImageCapture.CaptureMode.MIN_LATENCY)
        .setTargetRotation(getWindowManager().getDefaultDisplay().getRotation()).build();
    ImageCapture imgCap = new ImageCapture(imageCaptureConfig);

    ImageAnalysisConfig config =
        new ImageAnalysisConfig.Builder()
            .setTargetResolution(new Size(1280, 720))

.setImageReaderMode(ImageAnalysis.ImageReaderMode.ACQUIRE_LATEST_IMAGE)
        .build();

    ImageAnalysis imageAnalysis = new ImageAnalysis(config);

    imageAnalysis.setAnalyzer(
        new ImageAnalysis.Analyzer() {
            @Override
            public void analyze(ImageProxy image, int rotationDegrees) {
                myAnalyzer.analyze(image, rotationDegrees);
            }
        });
    CameraX.bindToLifecycle(this, preview, imgCap, imageAnalysis);
}

```

Tabela 3 - Excerto de classe privada para análise das imagem capturadas

```

private class YourAnalyzer implements ImageAnalysis.Analyzer {

    private int degreesToFirebaseRotation(int degrees) {
        switch (degrees) {
            case 0:
                return FirebaseVisionImageMetadata.ROTATION_0;
            case 90:
                return FirebaseVisionImageMetadata.ROTATION_90;
            case 180:
                return FirebaseVisionImageMetadata.ROTATION_180;
            case 270:
                return FirebaseVisionImageMetadata.ROTATION_270;
        }
    }
}

```

 <p>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</p>	<p>Computação Móvel e Ubíqua</p> <p>1º Semestre</p> <p>Docente: fas</p> <p>Ficha Prática: Firebase Libraries</p>
---	---

```

        default:
            throw new IllegalArgumentException(
                "Rotation must be 0, 90, 180, or 270.");
    }
}

@Override
public void analyze(ImageProxy imageProxy, int degrees) {
    if (imageProxy == null || imageProxy.getImage() == null) {
        return;
    }
    Image mediaImage = imageProxy.getImage();
    int rotation = degreesToFirebaseRotation(degrees);
    FirebaseVisionImage image =
        FirebaseVisionImage.fromMediaImage(mediaImage, rotation);
    // Pass image to an ML Kit Vision API
    // ...
    scanBarcodes(image);
}
}

```