배우기 firebase-authentication

#firebase-authentication

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: firebase-authentication

It is an unofficial and free firebase-authentication ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official firebase-authentication.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: firebase-authentication

firebase-authentication , .

firebase . firebase-authentication , .

Examples

Firebase v3 . Facebook, Github, Google, Twitter, . Github . .

. . .

, , Firebase Realtime Database Security Rules .

1.
   1. Firebase . *!*
   2. IDE. IDE ? Cloud9 . *!*
   3. Github, Google, Facebook Twitter . *!*
   4. 2 . *!*

2. **IDE**

   2. HTML5 .
   3. Firebase Tools . `npm install -g firebase-tools`
   4. Firebase Tools Firebase . `firebase login --no-localhost`
   5. Firebase Tools Firebase . `firebase init`
   6. IDE . `git clone https://github.com/rhroyston/firebase-auth.git`
   7. Firebase Tools IDE Firebase . `firebase deploy`
   8. Firebase . ? .
   9. JS, CSS HTML `href` , `src` `background: url` .
      1. `href` `src` .
      2. .
      3. script.js 781 `privateLink.href = "../firebase-auth/private"` .. .
      4. Firebase Hosting ( ) .

3. **Firebase**

   1. 6 . .
   2. Firebird URL + '/ ack'(: `https://my-app-1234/ack` Email Action Handler URL .

4.
   1. oAuth .
   2. `demo.update('mynode','myKey','myValue')` .
      1. .
      2. `href` .

```
demo.update("markup","secureData","<div class=\"mdl-card__title\"> <h1 class=\"mdl-
card__title-text mdl-color-text--white\">Secured Data</h1> </div><div class=\"mdl-
card__supporting-text mdl-typography--headline\"> <p>This is a secure card. The HTML
markup that renders this card is secured in the Realtime Database.  Access is determined
server side so no matter what you do with JavaScript on your browser you will not be able
to view this card unless you are authorized to.</p><p>Secured data can be markup, JSON,
strings, numbers, etc. Your imagination is the limit!</p></div><div class=\"mdl-
card__actions mdl-card--border intro-card-actions\"> <a class=\"mdl-button mdl-button--
colored mdl-js-button mdl-js-ripple-effect\" href=\"../firebase-auth/\">Home</a></div>");
```

Firebase v3

1. **oAuth**
    1. .
    2. .
    3. . ?
2.
    1. ?
    2. ?
    3. ?
    4. ?
    5. revokation ?
3.
    1. ?
    2. ?
    3. ?

Firebase Authentication Firebase .

Android .

, :

1. JavaScript Firebase .
2. Firebase Firebase .
3. / : Firebase Auth . / .

displayName, .createUserWithEmailAndPassword .updateProfile . .

```
function registerPasswordUser(email,displayName,password,photoURL){
  var user = null;
  //NULLIFY EMPTY ARGUMENTS
  for (var i = 0; i < arguments.length; i++) {
    arguments[i] = arguments[i] ? arguments[i] : null;
  }
  auth.createUserWithEmailAndPassword(email, password)
  .then(function () {
    user = auth.currentUser;
    user.sendEmailVerification();
  })
  .then(function () {
    user.updateProfile({
      displayName: displayName,
      photoURL: photoURL
    });
  })
  .catch(function(error) {
    console.log(error.message,7000);
  });
  console.log('Validation link was sent to ' + email + '.');
}
```

firebase-authentication   : https://riptutorial.com/ko/firebase-authentication/topic/4744/firebase-authentication-

2: Android Google Plus

Examples

Google Plus

Plus

onCreate

```
 GoogleSignInOptions gso = new
GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN)
 .requestIdToken(getString(R.string.default_web_client_id))
 .requestScopes(new Scope(Scopes.PLUS_LOGIN))

 .requestEmail()
 .build();
  mGoogleApiClient = new GoogleApiClient.Builder(this)
 .enableAutoManage(this /* FragmentActivity */, this /* OnConnectionFailedListener */)
 .addConnectionCallbacks(this)
 .addOnConnectionFailedListener(this)
 .addApi(Auth.GOOGLE_SIGN_IN_API,gso)
 .addApi(Plus.API)
 .build();
```

onStart ()

```
protected void onStart() {
  super.onStart();
  mGoogleApiClient.connect();
  mAuth.addAuthStateListener(mAuthListener);
}


protected void onStop() {
  super.onStop();
  if (mAuthListener != null) {
    mAuth.removeAuthStateListener(mAuthListener);
  }
  if (mGoogleApiClient.isConnected()) {
    mGoogleApiClient.disconnect();
  }
}

@Override
public void onConnected(Bundle bundle) {
  mSignInClicked = false;
  getProfileInformation();
  //getGoogleOAuthTokenAndLogin();
  Toast.makeText(this, "User is connected! (in onConnected
MActivty)",Toast.LENGTH_LONG).show();
  // Get user's information
  // Update the UI after signin
  //  updateUI(true);
}
```

```
@Override
public void onConnectionSuspended(int i) {

  mGoogleApiClient.connect();
  //updateUI(false);
}

@Override
public void onConnectionFailed(ConnectionResult connectionResult) {

  if (!connectionResult.hasResolution()) {
    GooglePlayServicesUtil.getErrorDialog(connectionResult.getErrorCode(), this,
            0).show();
  } else if(connectionResult.hasResolution()) {
    mConnectionResult = connectionResult;
    resolveSignInError();
  }
}



private void resolveSignInError() {
  Log.e("pavan", "User RESOLVE SIGN IN ERROR CALLED OUT OF IF");
  if(mSignInClicked){
    if (mConnectionResult.hasResolution()) {
      try {
          Log.e("pavan", "User RESOLVE SIGN IN ERROR CALLED OUT OF IF  TRY");
          mIntentInProgress = true;
          mConnectionResult.startResolutionForResult(this, GOOGLE_SIGIN);
      } catch (IntentSender.SendIntentException e) {
          Log.e("pavan", "User RESOLVE SIGN IN ERROR CALLED OUT OF IF  CATCH");
          mIntentInProgress = false;
          mGoogleApiClient.connect();
      }
    }
  }
}
```

```
private void getProfileInformation() {
  try {
      if (Plus.PeopleApi.getCurrentPerson(mGoogleApiClient) != null) {
        Person currentPerson = Plus.PeopleApi
                .getCurrentPerson(mGoogleApiClient);
        String personName = currentPerson.getDisplayName();
        String personPhotoUrl = currentPerson.getImage().getUrl();
        String personGooglePlusProfile = currentPerson.getUrl();
        String email = Plus.AccountApi.getAccountName(mGoogleApiClient);

  } catch (Exception e) {
      e.printStackTrace();
  }
}
```

Firebase ,

```
private void firebaseAuthWithGoogle(GoogleSignInAccount acct) {
//AuthCredential credential = GoogleAuthProvider.getCredential(tokenCredential, null);
AuthCredential credential = GoogleAuthProvider.getCredential(acct.getIdToken(),null);
mAuth.signInWithCredential(credential)
        .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                Log.d("TAG", "signInWithCredential:onComplete:" + task.isSuccessful());


                // If sign in fails, display a message to the user. If sign in succeeds
                // the auth state listener will be notified and logic to handle the
                // signed in user can be handled in the listener.
                if (!task.isSuccessful()) {
                    Log.w("TAG", "signInWithCredential", task.getException());
                    Toast.makeText(MainActivity.this, "Authentication failed.",
                            Toast.LENGTH_SHORT).show();
                }else{
                    SharedPreferences prefs =
PreferenceManager.getDefaultSharedPreferences(getApplicationContext());
                    Log.e("SahajLOG", "Login PREF ISSSSSSSS ONACTIVITYRESULT
"+prefs.getBoolean("AuthByGplus", AuthByGplus));
                    prefs.edit().putBoolean("AuthByGplus", true).commit();
                    Log.e("SahajLOG", "Login PREF ISSSSSSSS ONACTIVITYRESULT IFTRUE..
"+prefs.getBoolean("AuthByGplus", AuthByGplus));
                    Intent intent=new Intent(getApplicationContext(),MainActivity.class);
                    startActivity(intent);
                    finish();
                }
                // [START_EXCLUDE]
                // hideProgressDialog();
                // [END_EXCLUDE]
            }
        });
}
```

onActivityResult

```
@Override
protected void onActivityResult(int requestCode, int responseCode,
                            Intent intent) {

if (requestCode == GOOGLE_SIGIN) {
    if (responseCode != RESULT_OK) {
        mSignInClicked = false;
    }
    mIntentInProgress = false;
    if (!mGoogleApiClient.isConnecting()) {
        mGoogleApiClient.connect();
        if (mGoogleApiClient.isConnected()) {
           // getGoogleOAuthTokenAndLogin();
        }
    }
    GoogleSignInResult result = Auth.GoogleSignInApi.getSignInResultFromIntent(intent);
    if (result.isSuccess()) {
        // Google Sign In was successful, authenticate with Firebase
        GoogleSignInAccount account = result.getSignInAccount();
        Log.e("SahajLOG", "account   " + account.getIdToken());
        //getGoogleOAuthTokenAndLogin();
        firebaseAuthWithGoogle(account);
    } else {
        // Google Sign In failed, update UI appropriately
        // [START_EXCLUDE]
```

```
       Log.e("SahajLOG", "SIGN IN FAILED");
        // [END_EXCLUDE]
    }

}




public void signInWithGplus() {
if (!mGoogleApiClient.isConnecting()) {
    mSignInClicked = true;
    Intent signInIntent = Auth.GoogleSignInApi.getSignInIntent(mGoogleApiClient);
    startActivityForResult(signInIntent, GOOGLE_SIGIN);

 }

}
```

```
@Override
public void onLogout() {
  mAuth.signOut();
  if (mGoogleApiClient.isConnected()) {
    Plus.AccountApi.clearDefaultAccount(mGoogleApiClient);
    mGoogleApiClient.disconnect();
    mGoogleApiClient.connect();
    SignedInWithGoogle=false;
  }
  // Google sign out

  switchToLoginFragment();
}
```

Android Google Plus   : https://riptutorial.com/ko/firebase-authentication/topic/8887/android--google-plus-

S. No		Contributors
1	firebase-authentication	Big Dude, Community, ErstwhileIII, Ron Royston
2	Android Google Plus	Sahaj Rana, ThunderStruct