

Mini Project Series #1

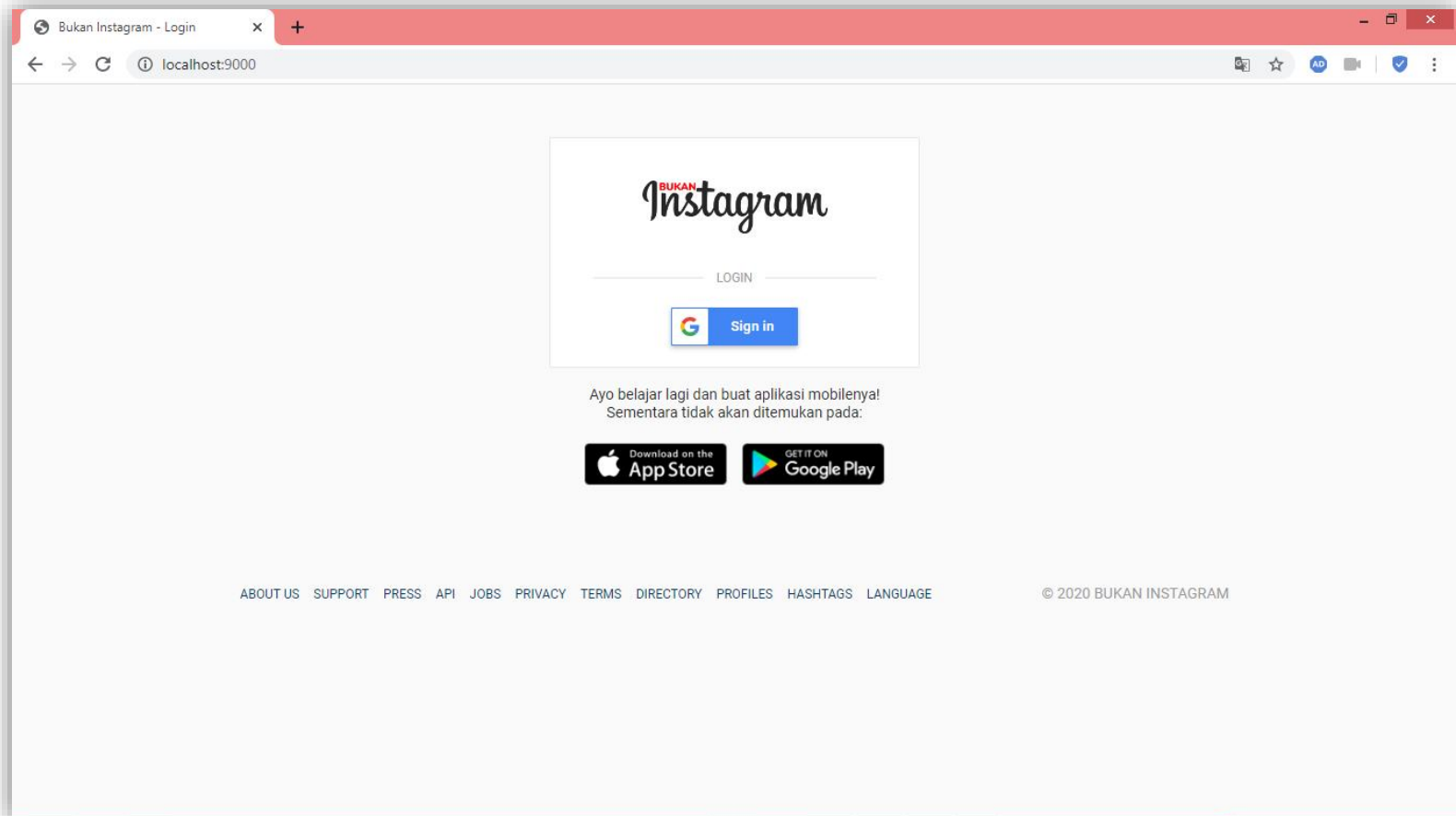
Login Screen

menggunakan Google OAuth



Arti Hikmatullah Perbawana Sakti Buana, ST
@saktibuana

Yang akan kita buat kali ini



Persiapan

- ❑ Kita akan memodifikasi 3 file.
Yaitu, index.html, style.css, dan script.js
- ❑ File gambar yang diperlukan sudah disediakan pada tautan berikut:
<https://drive.google.com/drive/folders/1mzxwJaK4d6NPz-94Mz2joxsAkNxBxBIG?usp=sharing>
- ❑ Buat dan simpanlah file-file ini di laptop/pc Anda dengan folder “bukan-Instagram”.
- ❑ Bukalah folder tersebut dengan *text editor* atau *IDE kesayangan* Anda.
- ❑ Install XAMPP (kita hanya memerlukan Apache HTTP Server nya saja)

Langkah 1:

Arahkan browser Google Chrome
Anda ke Google Developer Console
menggunakan alamat email google
Anda dan buatlah project baru.

<https://console.developers.google.com/>



Welcome Buana!

Create and manage your Google Cloud Platform instances, disks, networks, and other resources in one place.

Country

Indonesia

Terms of Service

☒ I agree to the [Google Cloud Platform Terms of Service](#), and the terms of service of [any applicable services and APIs](#).

Email updates

☒ I would like to receive periodic emails on news, product updates and special offers from Google Cloud and Google Cloud Partners.

[AGREE AND CONTINUE](#)

Langkah 2:

Pastikan sudah memberi nama dan id yang sesuai, lalu tekan tombol **CREATE**

<https://console.developers.google.com/>

Project name *


Mini Project

Project ID *

mini-project-281203

Project ID can have lowercase letters, digits, or hyphens. It must start with a lowercase letter and end with a letter or number.

Location *

 No organization [BROWSE](#)

Parent organization or folder

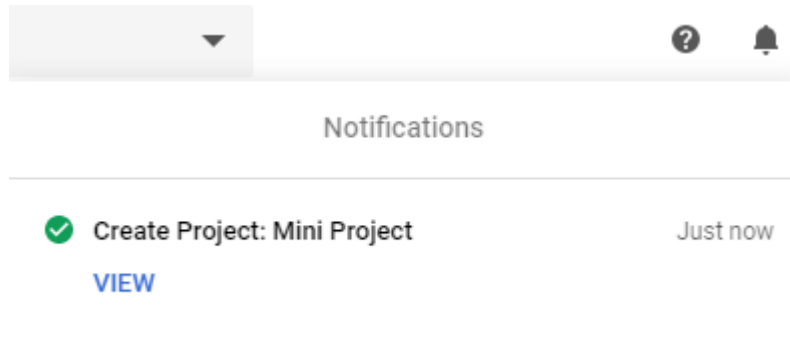
CREATE

CANCEL

Langkah 3:

Klik **VIEW** dari Project yang barusan kita buat

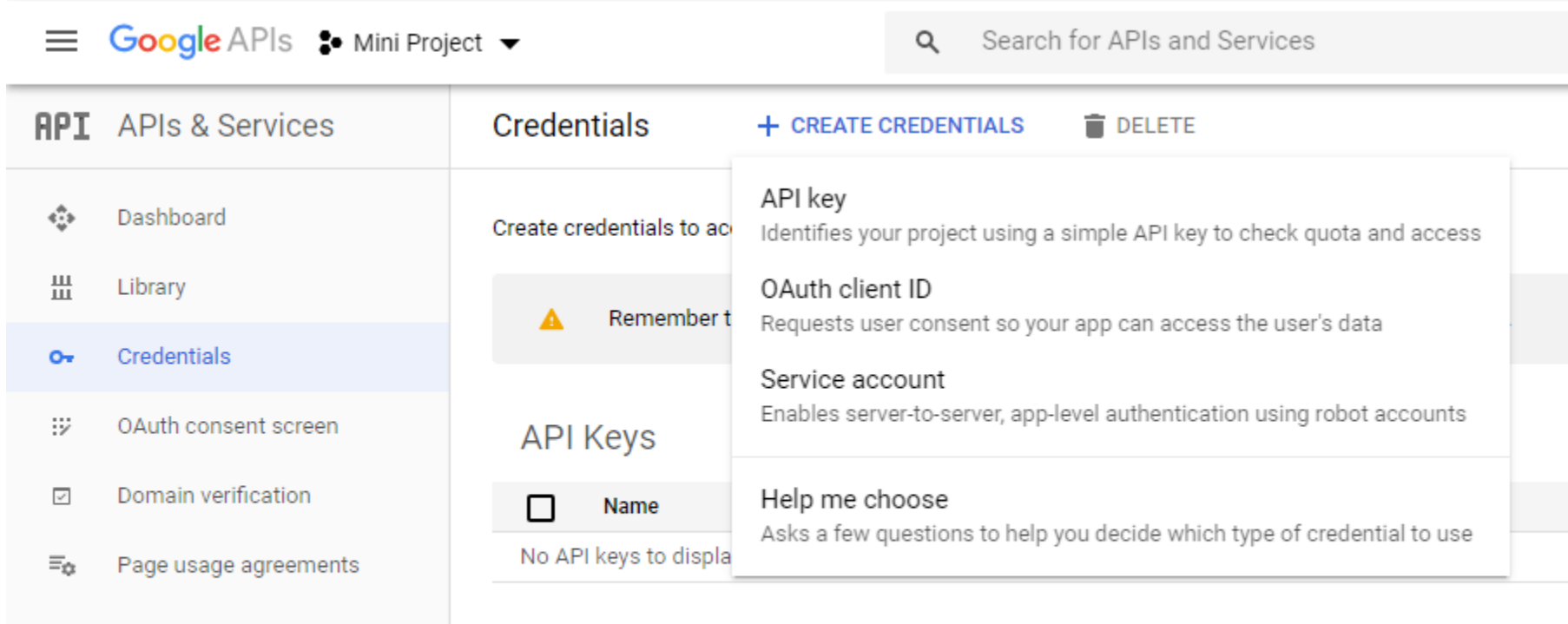
<https://console.developers.google.com/>



Langkah 4:

Dapatkanlah **Client ID** dengan cara menekan submenu **OAuth client ID** pada menu **Create Credentials**

<https://console.developers.google.com/>



Google APIs Mini Project

Search for APIs and Services

API APIs & Services

Credentials + CREATE CREDENTIALS DELETE

Create credentials to access your project's APIs

Remember to...

API Keys

No API keys to display

- API key**
Identifies your project using a simple API key to check quota and access
- OAuth client ID**
Requests user consent so your app can access the user's data
- Service account**
Enables server-to-server, app-level authentication using robot accounts
- Help me choose**
Asks a few questions to help you decide which type of credential to use

Dashboard

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Langkah 5:

[CONFIGURE CONSENT SCREEN](#)

Jika muncul tombol **CONFIGURE CONSENT SCREEN** tekanlah dan pastikan memilih **User Type: External** sebelum menekan tombol **CREATE**.

OAuth consent screen

Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.

User Type

☐ Internal ?

Only available to users within your organization. You will not need to submit your app for verification.

☒ External ?

Available to any user with a Google Account.

[CREATE](#)

Langkah 6:

Pada langkah ini yang harus (mandatory) untuk diisi adalah Application name dan Application logo. Setelah diisi maka segeralah tekan tombol **SAVE** pada area bawah halaman ini.

Note: Kita tidak harus menunggu status verified, dan bahkan tidak memerlukan status terverifikasi pada tahap ini.

OAuth consent screen

Before your users authenticate, this consent screen will allow them to choose whether they want to grant access to their private data, as well as give them a link to your terms of service and privacy policy. This page configures the consent screen for all applications in this project.

Verification status

☒ Needs verification

Application name [?]

The name of the app asking for consent

Application logo [?]

An image on the consent screen that will help users recognize your app




Langkah 7:

Jika pada langkah ini muncul verification status **Needs verification**, jangan khawatir karena ini justru ciri langkah sebelumnya sudahlah benar.

Kembali lah ke menu **Credentials**

Catatan: Sekali lagi, kita tidak harus menunggu status **verified**, dan bahkan tidak memerlukan status **verified** pada project ini.

OAuth consent screen

mini-project-bukan-instagram  EDIT APP

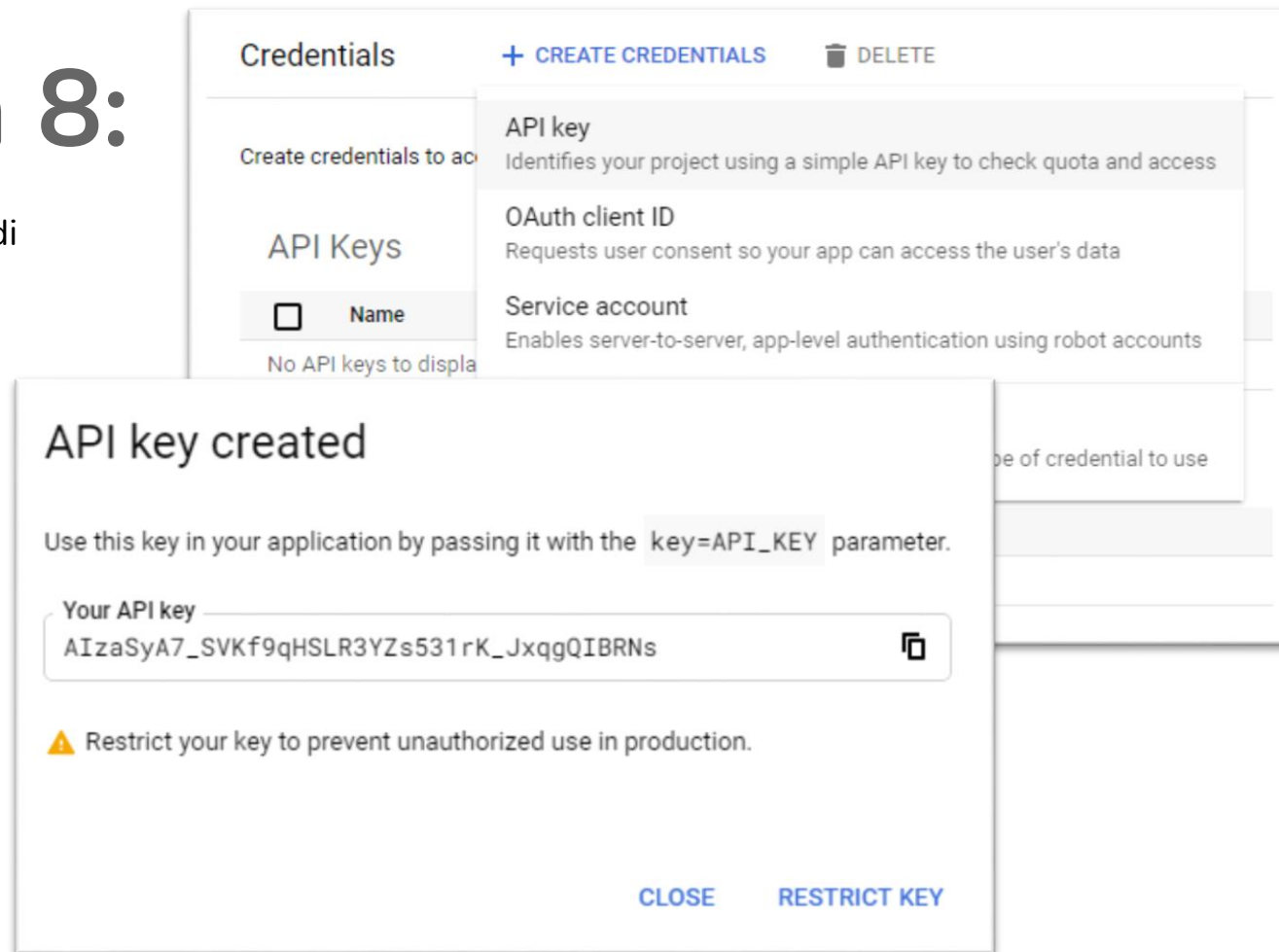
Verification Status

 Needs verification

Langkah 8:

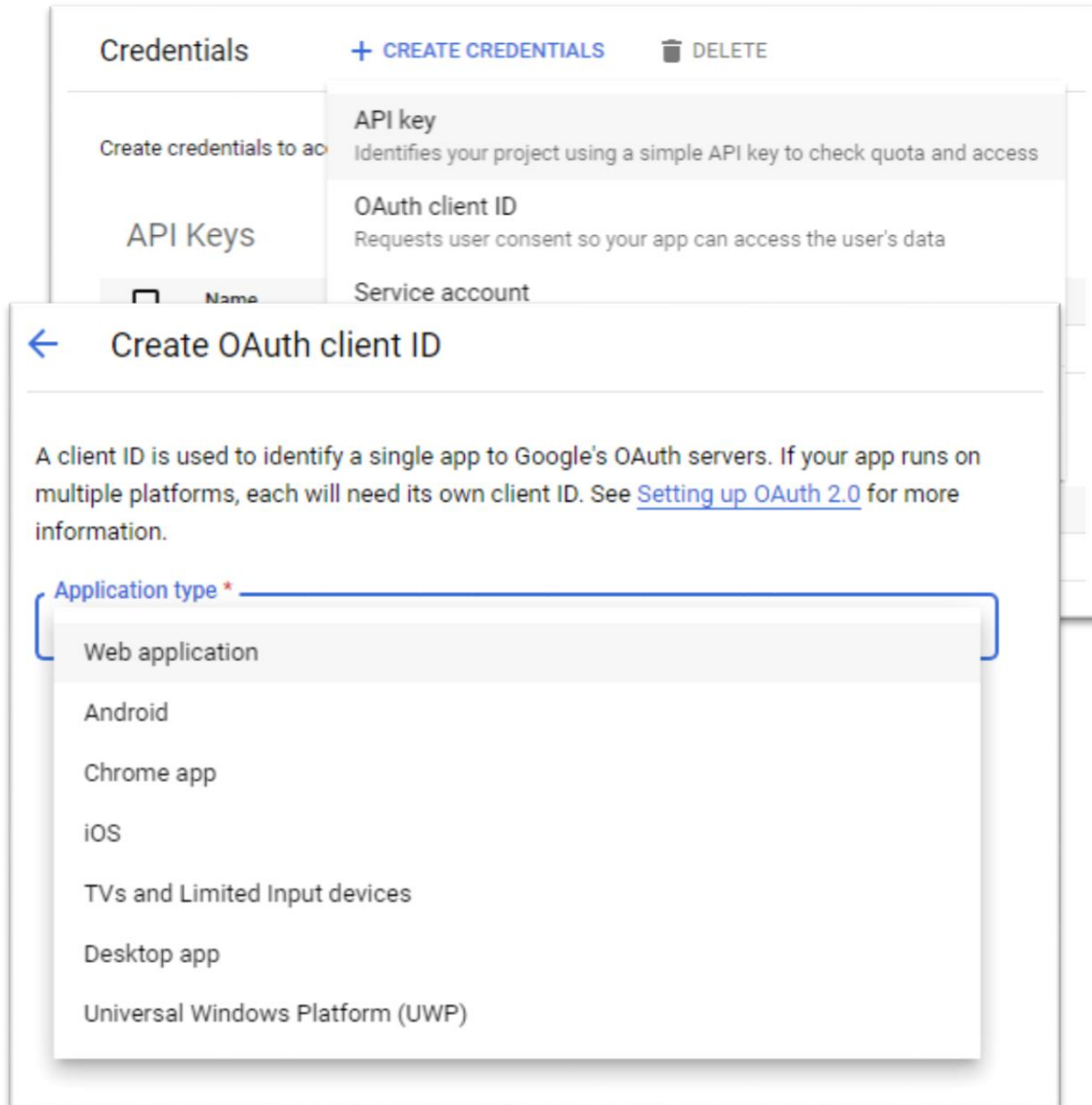
Pilihlah submenu **API Key** di dropdown menu **CREATE CREDENTIALS** dan pop up modal window akan memunculkan **API key**.

Copy, paste ditempat aman, lalu tekan tombol **close**.



Langkah 9:

Pilihlah submenu **Oauth Client ID** di dropdown menu **CREATE CREDENTIALS** dan Pilih opsi **Web application** pada dropdown **Application Type**



Langkah 10:

Pada tahap ini, yang mandatory adalah **Name** dan **URIs**, kita isikan dengan **mini-project-2** & <http://localhost:9000>, lalu tekan tombol **CREATE**

←

Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth service. If you have multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information.

Application type *

Web application

[Learn more](#) about OAuth client types

Name *

mini-project-2

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

←

Create OAuth client ID

Authorized JavaScript origins ?

For use with requests from a browser

URIs

http://localhost:9000

⌵

+ ADD URI

Authorized redirect URIs ?

For use with requests from a web server

URIs

http://localhost:9000

+ ADD URI

CREATE

CANCEL

Langkah 11:

Pop up modal window akan memunculkan OAuth Client ID.

Copy, paste ditempat aman, lalu tekan tombol OK.

OAuth client created

The client ID and secret can always be accessed from Credentials in APIs & Services



OAuth is limited to 100 [sensitive scope logins](#) until the [OAuth consent screen](#) is verified. This may require a verification process that can take several days.

Your Client ID

747385033609-furt64d64q3pobhsfn0dha6pamo5insn.apps.gc



Your Client Secret

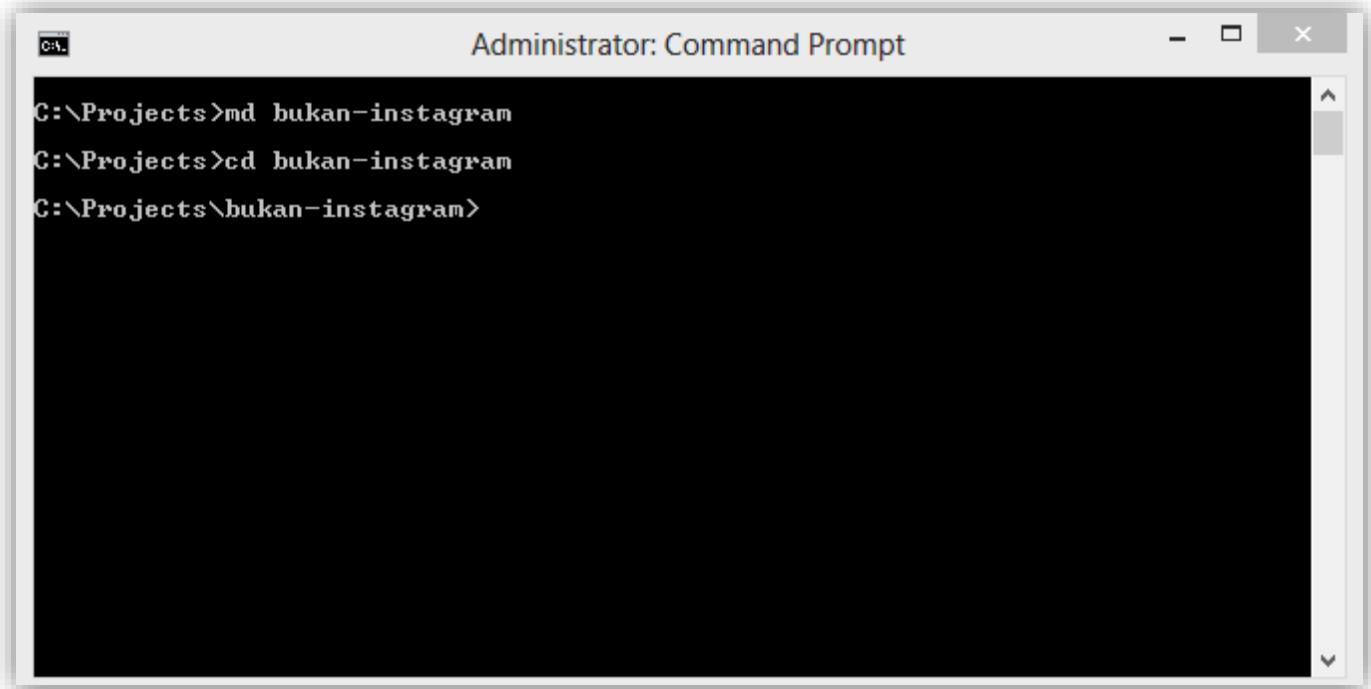
CwIGe65j_zZ0ehfouQ3APJfg



OK

Langkah 12:

1. Ketikkan perintah `md bukan-instagram` dan tekan tombol `Enter`, maka akan membuat folder bernama `bukan-instagram`
2. Ketikkan perintah `cd bukan-instagram` dan tekan tombol `Enter`, maka akan masuk kedalam direktori `bukan-instagram`

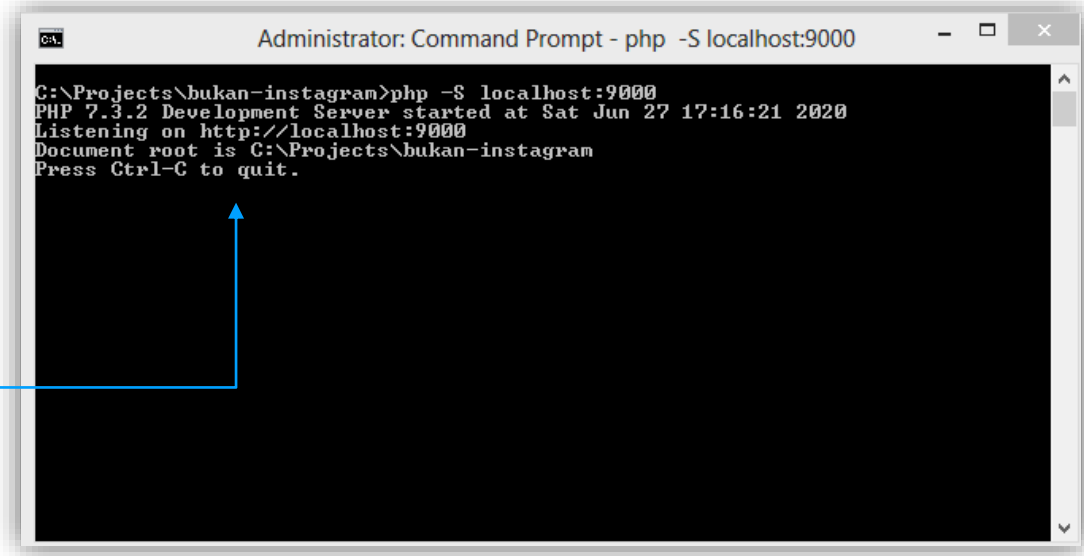


```
C:\>Administrator: Command Prompt

C:\>C:\Projects>md bukan-instagram
C:\Projects>cd bukan-instagram
C:\Projects\bukan-instagram>
```

Langkah 13:

Masih didalam direktori bukan-instagram, ketikkan perintah `php -S localhost:9000` dan tekan tombol Enter, maka Apache HTTP Server akan berjalan pada localhost dalam port 9000 dengan Document root `C:\Projects\bukan-instagram`



```
C:\Projects\bukan-instagram>php -S localhost:9000
PHP 7.3.2 Development Server started at Sat Jun 27 17:16:21 2020
Listening on http://localhost:9000
Document root is C:\Projects\bukan-instagram
Press Ctrl-C to quit.
```

Jika didapati tampilan seperti ini maka sudah pada jalan yang benar!

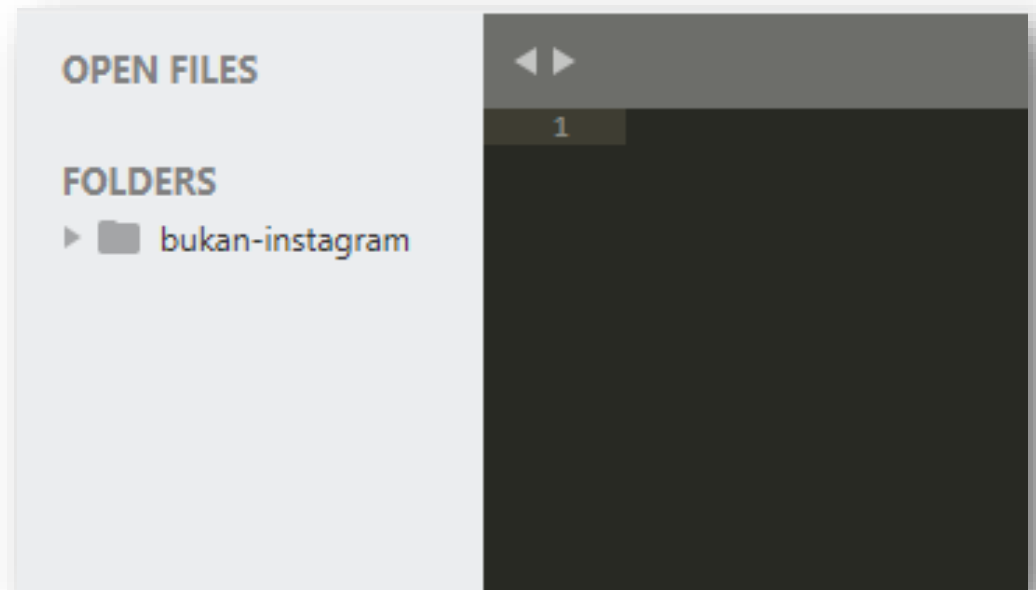
Jika belum menginstall HTTP Server, silahkan ikuti instruksi penginstallan pada tautan berikut ini:

https://drive.google.com/drive/folders/13xkQIFT2cS_7KMtRclvUgNpc4SJnbOtT

Langkah 14:

Sekarang kita siap melakukan pengetikan pada IDE Anda.

Buka folder “bukan-Instagram” Anda dan ikuti perintah selanjutnya.



Menambahkan struktur dasar HTML Document

- ❑ Tambahkan struktur dasar HTML ke index.html.
- ❑ Deklarasi DOCTYPE, <html> tag, <head> tag dan <body> diperlukan.

rf. <https://progate.com/html/study/1/10#/30>

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4
5      </head>
6      <body>
7
8      </body>
9  </html>
```

Tambahkan konten ke dalam <head> Tag

- ❑ Tambahkan konten ke dalam <head> tag. Character Encoding, Judul dan link ke file CSS diperlukan.

rf. <https://progate.com/html/study/1/11#/33>

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Bukan Instagram - Login</title>
  <link rel="stylesheet" href="static/css/styles.css">
</head>
```

Layout Utama Halaman Login

BUKAN
Instagram

LOGIN



Ayo belajar lagi dan buat aplikasi mobilenya!
Sementara tidak akan ditemukan pada:



[ABOUT US](#) [SUPPORT](#) [PRESS](#) [API](#) [JOBS](#) [PRIVACY](#) [TERMS](#) [DIRECTORY](#) [PROFILES](#) [HASHTAGS](#) [LANGUAGE](#)

© 2020 BUKAN INSTAGRAM

Tambahan tags untuk <head></head> pada Halaman Login

- ❑ Tambahkan 2 tag <meta>
- ❑ Untuk <meta> pertama, sertakan attribute name berisikan google-signin-scope dan attribute content berisikan profile email.
- ❑ Untuk <meta> pertama, sertakan attribute name berisikan google-signin-client_id dan attribute content berisikan OAuth Client ID.
- ❑ Tambahkan tag <script> dengan attribute src berisikan <https://apis.google.com/js/platform.js> dan attribute async defer

```
<meta name="google-signin-scope" content="profile email">  
<meta name="google-signin-client_id" content="747385033609-furt64d64q3pobhsfn0dha6pamo5insn.apps.googleusercontent.com">  
<script src="https://apis.google.com/js/platform.js" async defer></script>
```

HTML untuk kerangka didalam <body></body>

- Tambahkan tag <div> dengan class “wrapper” dan 2 tag <div> didalamnya berisikan class “header” dan “footer”

```
1 <div class="wrapper">
2   <div class="header">
3
4   </div>
5   <div class="footer">
6
7   </div>
8 </div>
```

Tambahkan HTML ini untuk <div class="header">

- ❑ Lengkapi didalamnya dengan tag <div> dengan class "top" serta class "apps"

```
<div class="top">
  <div class="logo">
    
  </div>
  <div class="or">
    <div class="line"></div>
    <p>LOGIN</p>
    <div class="line"></div>
  </div>
  <div class="dif">
    <div class="gs">
      <div class="g-signin2" data-onsuccess="onSignIn" data-theme="dark"></div>
    </div>
    <div class="or logged onlylogged">
      <div class="line"></div>
      <p><a href="#" onclick="gotoDashboard();">Dashboard</a></p>
      <div class="line"></div>
    </div>
    <div class="or logged onlylogged">
      <div class="line"></div>
      <p><a href="#" onclick="onSignOut();">Sign out</a></p>
      <div class="line"></div>
    </div>
  </div>
</div>
<div>&nbsp;</div>
<div class="apps">
  <p>Ayo belajar lagi dan buat aplikasi mobilenya!<br/>Sementara tidak akan ditemukan pada:</p>
  <div class="icons">
    <a href="#"></a>
    <a href="#"></a>
  </div>
</div>
```

Tambahkan HTML untuk <div class="footer">

- ❑ Lengkapi didalamnya dengan tag <div> dengan class "links" serta class "copyright"

```
<div class="links">
  <ul>
    <li><a href="#">ABOUT US</a></li>
    <li><a href="#">SUPPORT</a></li>
    <li><a href="#">PRESS</a></li>
    <li><a href="#">API</a></li>
    <li><a href="#">JOBS</a></li>
    <li><a href="#">PRIVACY</a></li>
    <li><a href="#">TERMS</a></li>
    <li><a href="#">DIRECTORY</a></li>
    <li><a href="#">PROFILES</a></li>
    <li><a href="#">HASHTAGS</a></li>
    <li><a href="#">LANGUAGE</a></li>
  </ul>
</div>
<div class="copyright">
  © 2020 BUKAN INSTAGRAM
</div>
```


❑ Jika tampilan layar Anda sudah seperti ini berarti Anda sudah benar! Lanjutkan dengan menambahkan style.css!

Instagram

LOGIN

[Dashboard](#)

[Sign out](#)

Dapatkan aplikasinya!



- [ABOUT US](#)
- [SUPPORT](#)
- [PRESS](#)
- [API](#)
- [JOBS](#)
- [PRIVACY](#)
- [TERMS](#)
- [DIRECTORY](#)
- [PROFILES](#)
- [HASHTAGS](#)
- [LANGUAGE](#)

Menghubungkan HTML dengan CSS

- ❑ Gunakan tag syntax `<link rel="stylesheet" href="namafile.css" />`
- ❑ Perhatikan letak file yang akan kita hubungkan.

```
<link rel="stylesheet" href="static/css/styles.css">
</head>
<body>
```

- ❑ Tempatkan di baris sebelum tag `</head>`

Tambahkan CSS pada style.css

- ❑ Tambahkan `@import url('nama_url')` diawal baris,
- ❑ Gantikan `nama_url` dengan `https://fonts.googleapis.com/css?family=Roboto&display=swap`

```
@import url('https://fonts.googleapis.com/css?family=Roboto&display=swap');
```

- ❑ Tambahkan `scope *` dan gunakan font yang diinginkan dengan menuliskan “font-family”, serta atur attribute lain seperti pada gambar.

```
*{  
  margin: 0;  
  padding: 0;  
  box-sizing: border-box;  
  outline: none;  
  list-style: none;  
  text-decoration: none;  
  font-family: 'Roboto', sans-serif;  
}
```

Tambahkan CSS pada style.css

- ❑ Tambahkan scope body dengan attribute background: #fafafa; dan font-size: 14px;

```
body{  
  background: #fafafa;  
  font-size: 14px;  
}
```

- ❑ Tambahkan scope .wrapper yang merujuk pada class wrapper seperti pada gambar.

```
.wrapper .header{  
  max-width: 350px;  
  width: 100%;  
  height: auto;  
  margin: 50px auto;  
}  
  
.wrapper .header .top,  
.wrapper .signup{  
  background: #fff;  
  border: 1px solid #e6e6e6;  
  border-radius: 1px;  
  padding: 40px 40px 20px;  
}  
  
.wrapper .header .logo img{  
  display: block;  
  margin: 0 auto 35px;  
}  
  
.wrapper .header .form .input_field{  
  margin-bottom: 5px;  
}
```

Tambahkan CSS pada style.css

- ❑ Tambahkan lagi didalam selector class wrapper

```
.wrapper .header .form .input_field .input{
  width: 100%;
  background: #fafafa;
  border: 1px solid #efefef;
  font-size: 12px;
  border-radius: 3px;
  color: #262626;
  padding: 10px;
}

.wrapper .header .form .input_field .input:focus{
  border: 1px solid #b2b2b2;
}

.wrapper .header .form .btn{
  margin: 10px 0;
  background-color: #3897f0;
  border: 1px solid #3897f0;
  border-radius: 4px;
  text-align: center;
  padding: 5px;
}

.wrapper .header .form .btn a{
  color: #fff;
  display: block;
}

.wrapper .header .or{
  display: flex;
  justify-content: space-between;
  align-items: center;
  height: 15px;
  margin: 15px 0 20px;
}

.wrapper .header .or .line{
  width: 105px;
  height: 2px;
  background: #efefef
}

.wrapper .header .or p{
  color: #999;
  font-size: 12px;
}
```

```
.wrapper .dif .gs{
  display: flex;
  justify-content: center;
  align-items: center;
}

.wrapper .dif .forgot{
  font-size: 12px;
  text-align: center;
  margin-top: 20px;
}

.wrapper .dif .forgot a{
  color: #003569;
}

.wrapper .signup{
  margin: 10px 0 20px;
  padding: 25px 40px;
  text-align: center;
  color: #262626;
}

.wrapper .signup a{
  color: #3897f0;
}

.wrapper .apps{
  text-align: center;
  color: #262626;
}

.wrapper .apps p{
  margin-bottom: 20px;
}

.wrapper .apps a img{
  width: 135px;
  height: 40px;
  margin: 0 5px;
}
```

Tambahkan CSS pada style.css

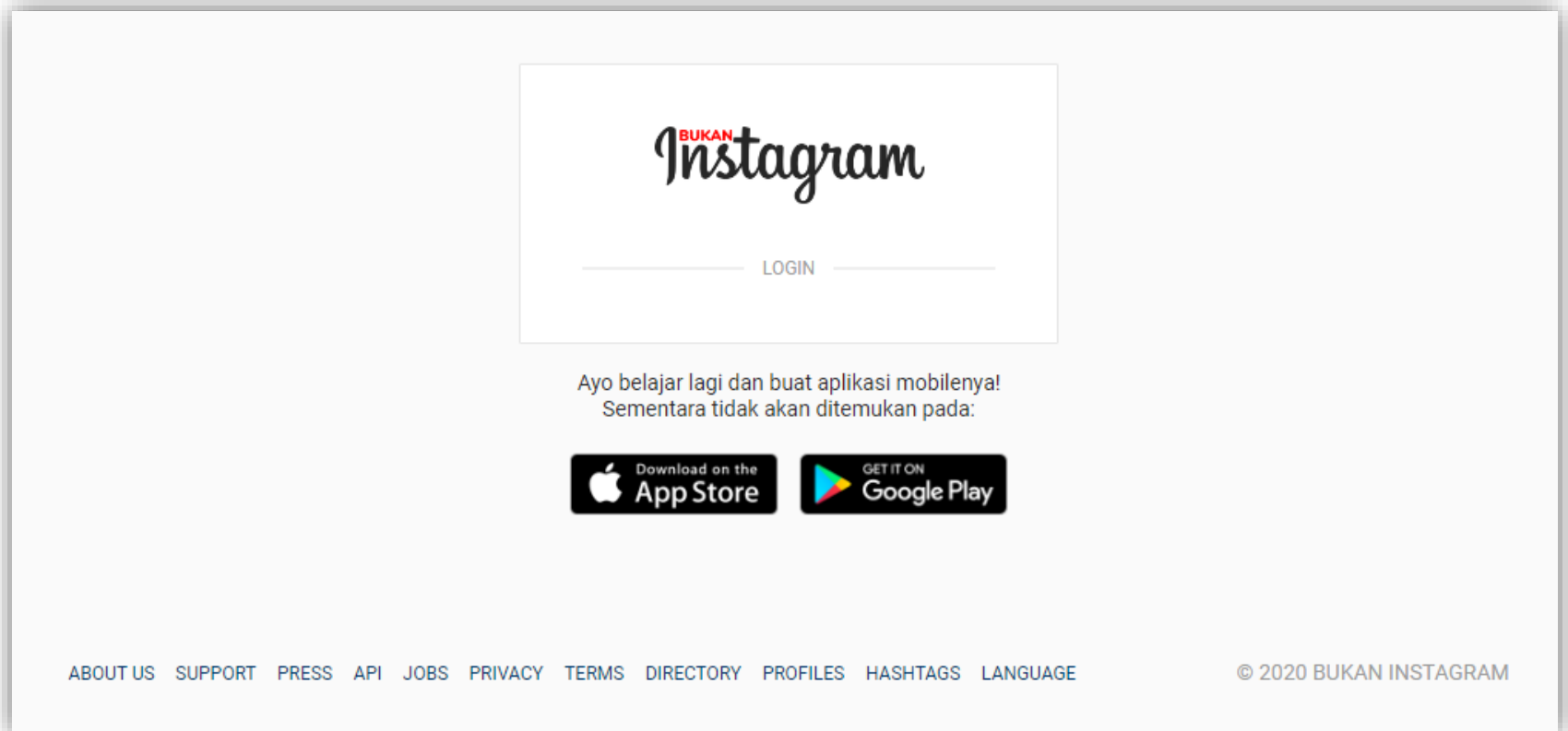
- ❑ Tambahkan lagi didalam selector class footer
- ❑ Terakhir, untuk menyembunyikan kondisi sebelum login, tambahkan selector class logged

```
.logged{  
    display:none !important;  
}
```

```
.footer{  
    max-width: 935px;  
    width: 100%;  
    margin: 0 auto;  
    padding: 40px 0;  
    display: flex;  
    justify-content: space-between;  
}  
  
.footer .links ul li{  
    display: inline-block;  
    margin-right: 10px;  
}  
  
.footer .links ul li a{  
    color: #003569;  
    font-size: 12px;  
}  
  
.footer .copyright{  
    color: #999;  
}
```

Periksa layoutnya

- ❑ Buka browser Anda di url → `http://localhost:9000`
- ❑ Jika tampilan layar Anda sudah seperti ini berarti Anda sudah benar! Lanjutkan!



Menghubungkan HTML dengan JavaScript

- ❑ Gunakan tag `<script>` dengan attribute `src` berisikan *path* menuju file berekstensi `js`.
- ❑ Perhatikanlah lokasi file `js`, hati hati gunakan `/` untuk membedakan folder dengan file
- ❑ Meskipun bisa ditempatkan dimana saja, namun alangkah lebih baik ditempatkan pada sebelum `</body>` agar elemen HTML lain sudah ter *render* oleh browser terlebih dahulu, sehingga siap untuk kita terapkan operasi logis pada `js`

```
71      <script src="static/js/login.js"></script>
72    </body>
73  </html>
```


Coding JS untuk “login.js”

❑ Tambahkan 4 buah *function* javascript berikut ini:

1. onSignIn ()
2. onSignOut ()
3. hideDashboard ()
4. gotoDashboard ()

❑ Pastikan pada *function* onSignIn diberikan parameter googleUser

❑ Perhatikan abstraksi pada gambar disamping

```
function onSignIn(googleUser) {  
      
}  
  
function onSignOut() {  
      
}  
  
function hideDashboard(){  
      
}  
  
function gotoDashboard(){  
      
}
```

Coding untuk file “login.js”

- ❑ Lengkapi logika pada function onSignIn() dan onSignOut()
- ❑ Pastikan juga sudah ada function hideDashboard() serta gotoDashboard()

```
function onSignIn(googleUser) {  
    var profile = googleUser.getBasicProfile();  
    console.log('Full Name: ' + profile.getName());  
    console.log("Email: " + profile.getEmail());  
  
    var x = document.getElementsByClassName("logged");  
    var i;  
    for (i = 0; i <= x.length; i++) {  
        x[0].classList.remove("logged");  
    }  
}
```

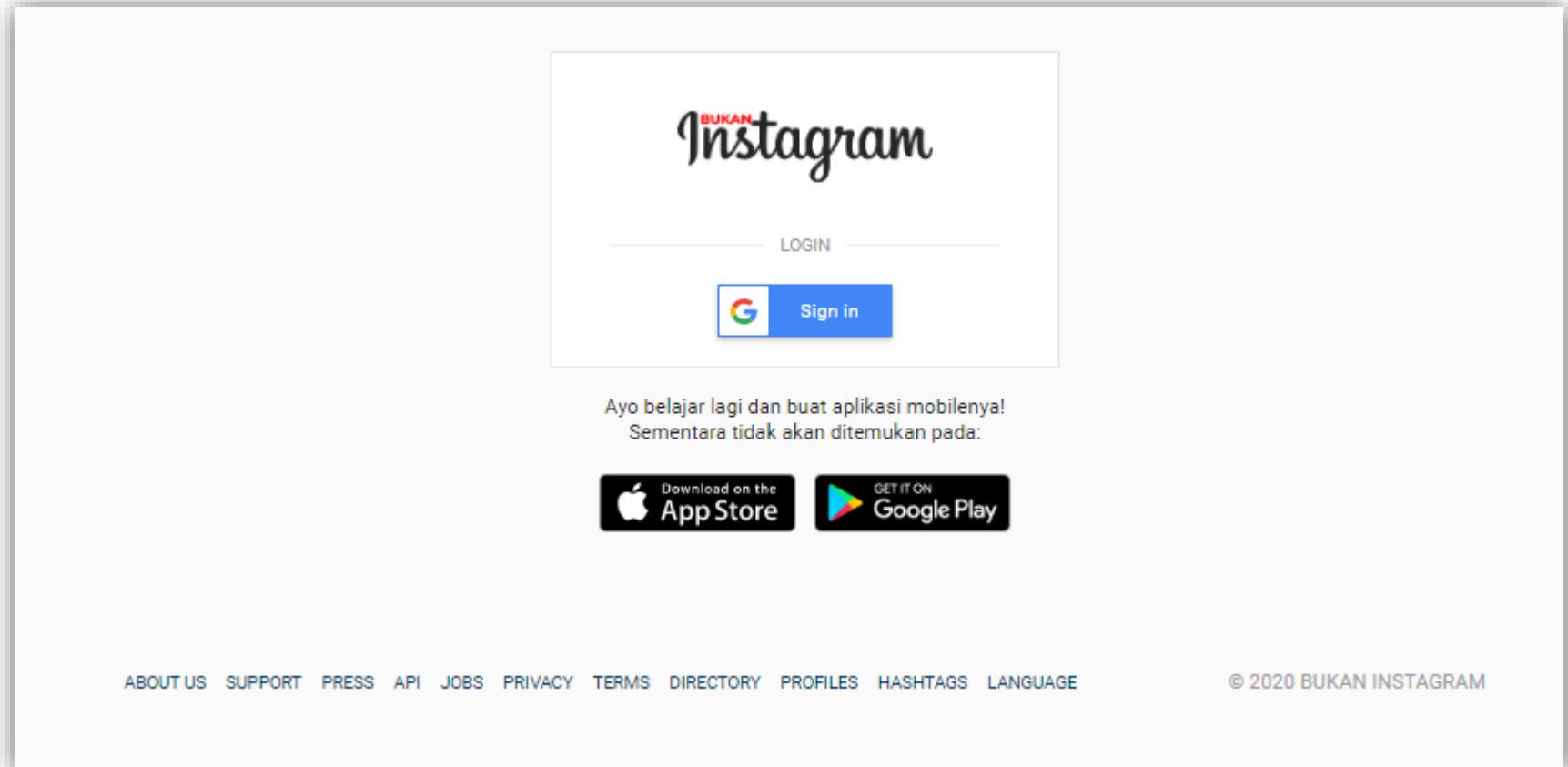
```
function onSignOut() {  
    var auth2 = gapi.auth2.getAuthInstance();  
    auth2.signOut().then(function () {  
        console.log('User signed out.');        hideDashboard();  
    });  
}
```

```
function hideDashboard(){  
    var x = document.getElementsByClassName("onlylogged");  
    var i;  
    for (i = 0; i <= x.length; i++) {  
        x[i].classList.add("logged");  
    }  
}
```

```
function gotoDashboard(){  
    window.location.replace("http://localhost:9000/dashboard.html");  
}
```

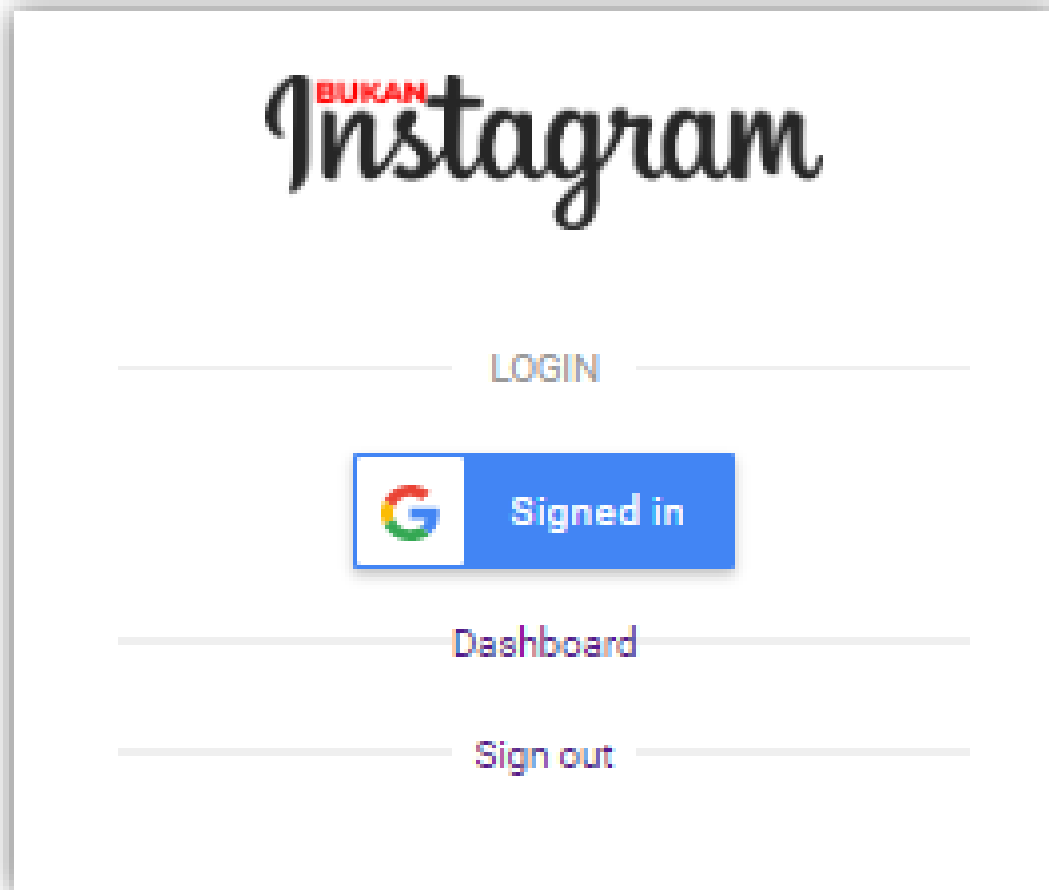
Periksa layoutnya

- ❑ Buka kembali browser di `http://localhost:9000`
- ❑ Jika tampilan layar Anda sudah seperti ini berarti Anda sudah benar! Selamat!



Cek fungsi Google OAuth

- ❑ Jika setelah dicoba menekan tombol google dan memasukkan user password gmail Anda berhasil dan tampak sudah seperti ini berarti Anda sudah benar! Selamat!



KITA BERHASIL!!!!



What's Next?



WHAT'S NEXT?

Yang dinanti pada mini project series selanjutnya ...

