

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## **Aim : Securing a web application with single sign-on using NodeJS.**

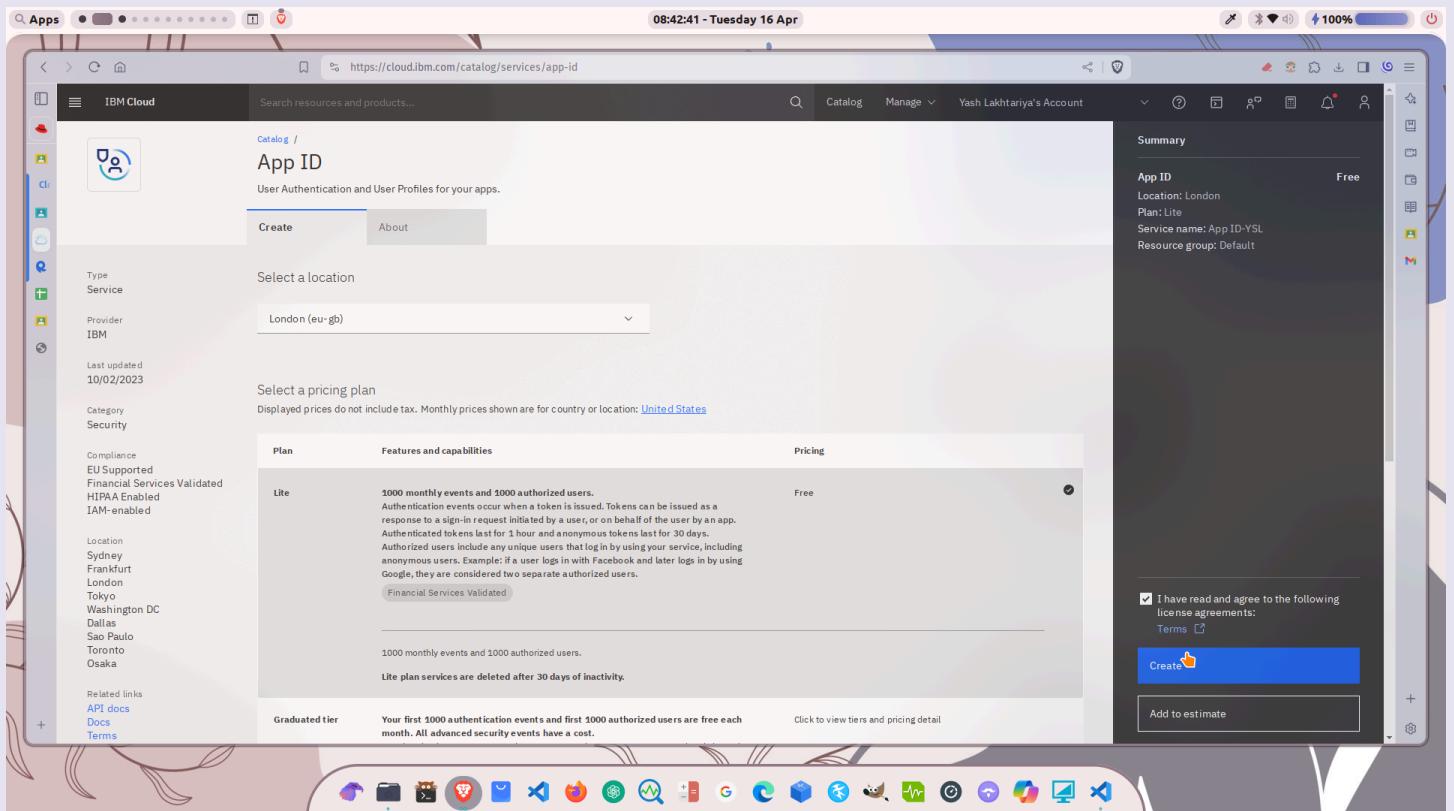
### **1. Create AppID service instance through IBM cloud**

### **2. Customize the AppID instance according to requirements**

### **3. Bind application with AppID service.**

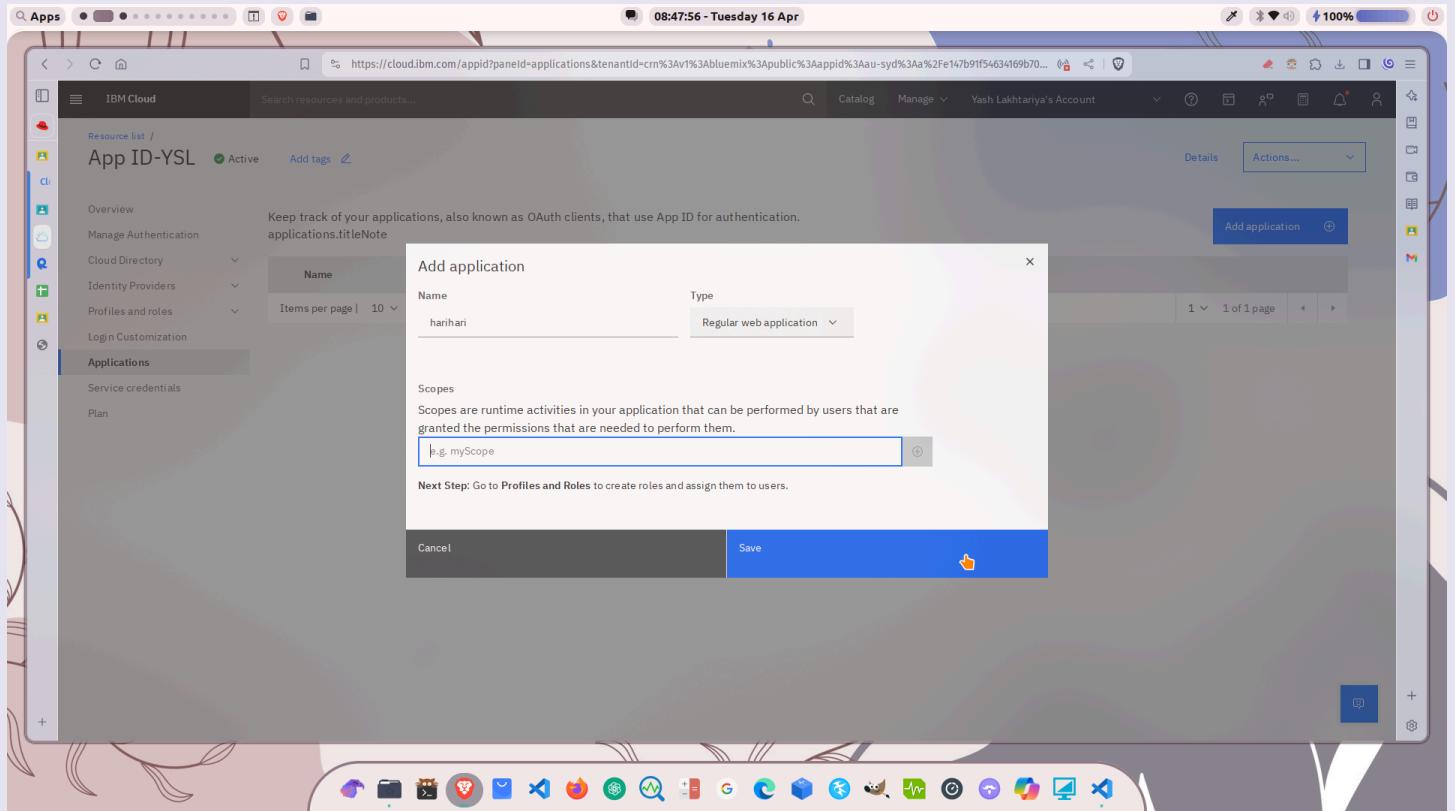
#### **Steps and Screenshots :**

##### **1. Create AppID instance in IBM Cloud**



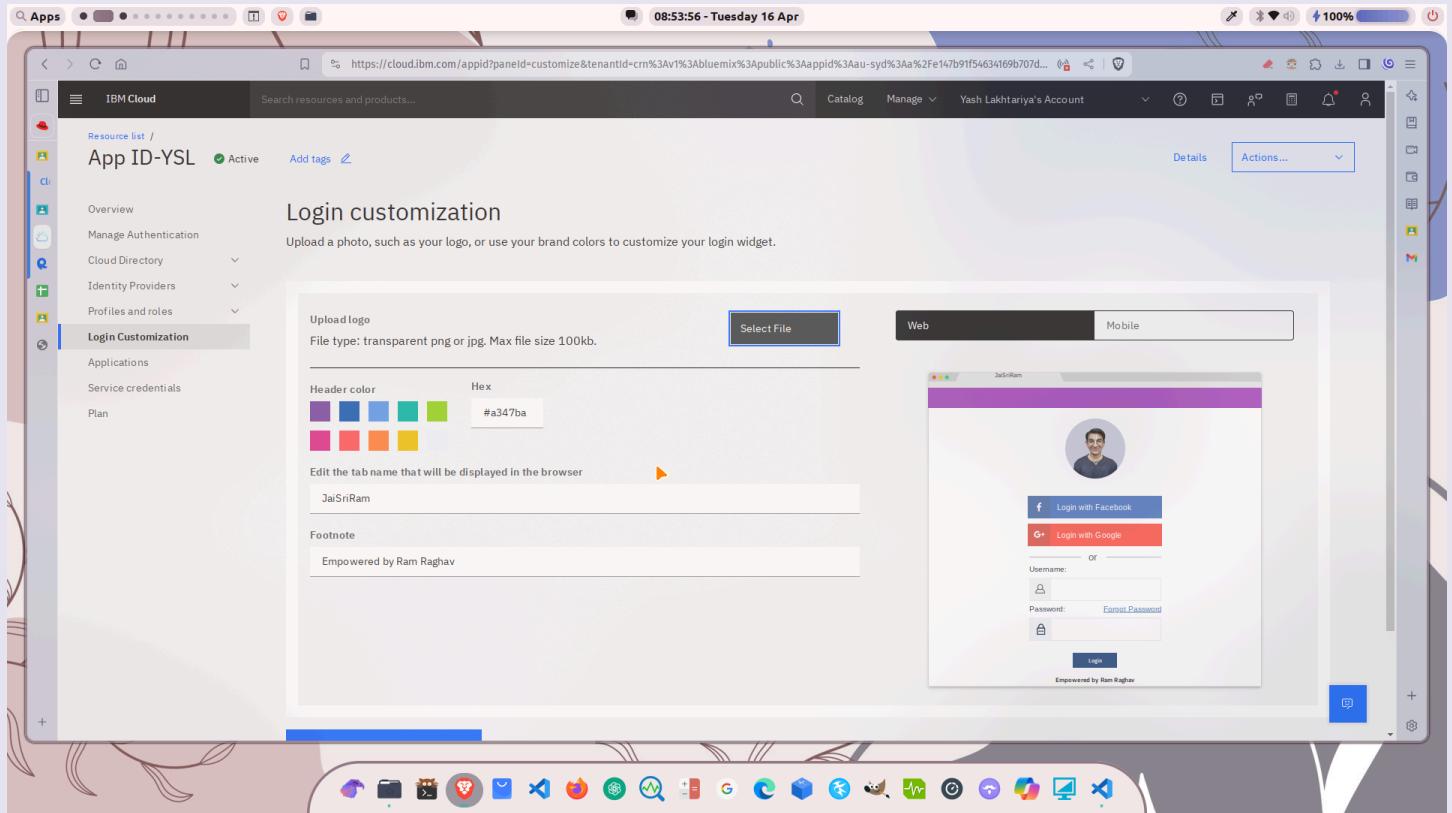
**Name - Yash Lakhtariya**  
**Enrollment number - 21162101012**  
**Branch - CBA      Batch - 61**  
**EADC Practical 15**

## 2. Create an application for regular web app



**Name - Yash Lakhtariya**  
**Enrollment number - 21162101012**  
**Branch - CBA      Batch - 61**  
**EADC Practical 15**

### 3. Customize the appearance of login page using Login Customization



The screenshot shows the IBM Cloud App ID customization interface. On the left, a sidebar lists various app settings like Overview, Manage Authentication, Cloud Directory, Identity Providers, Profiles and roles, and Login Customization (which is currently selected). The main area is titled "Login customization" and contains fields for "Upload logo" (with a "Select File" button), "Header color" (a color palette and hex code #a347ba), "Tab name" (set to "JaiSriRam"), and "Footnote" (set to "Empowered by Ram Raghav"). To the right, there are tabs for "Web" and "Mobile", and a preview window showing a purple-themed login page with a user profile picture, social login buttons for Facebook and Google, and standard login fields for Username and Password. The bottom of the screen shows a Mac OS X dock with various application icons.

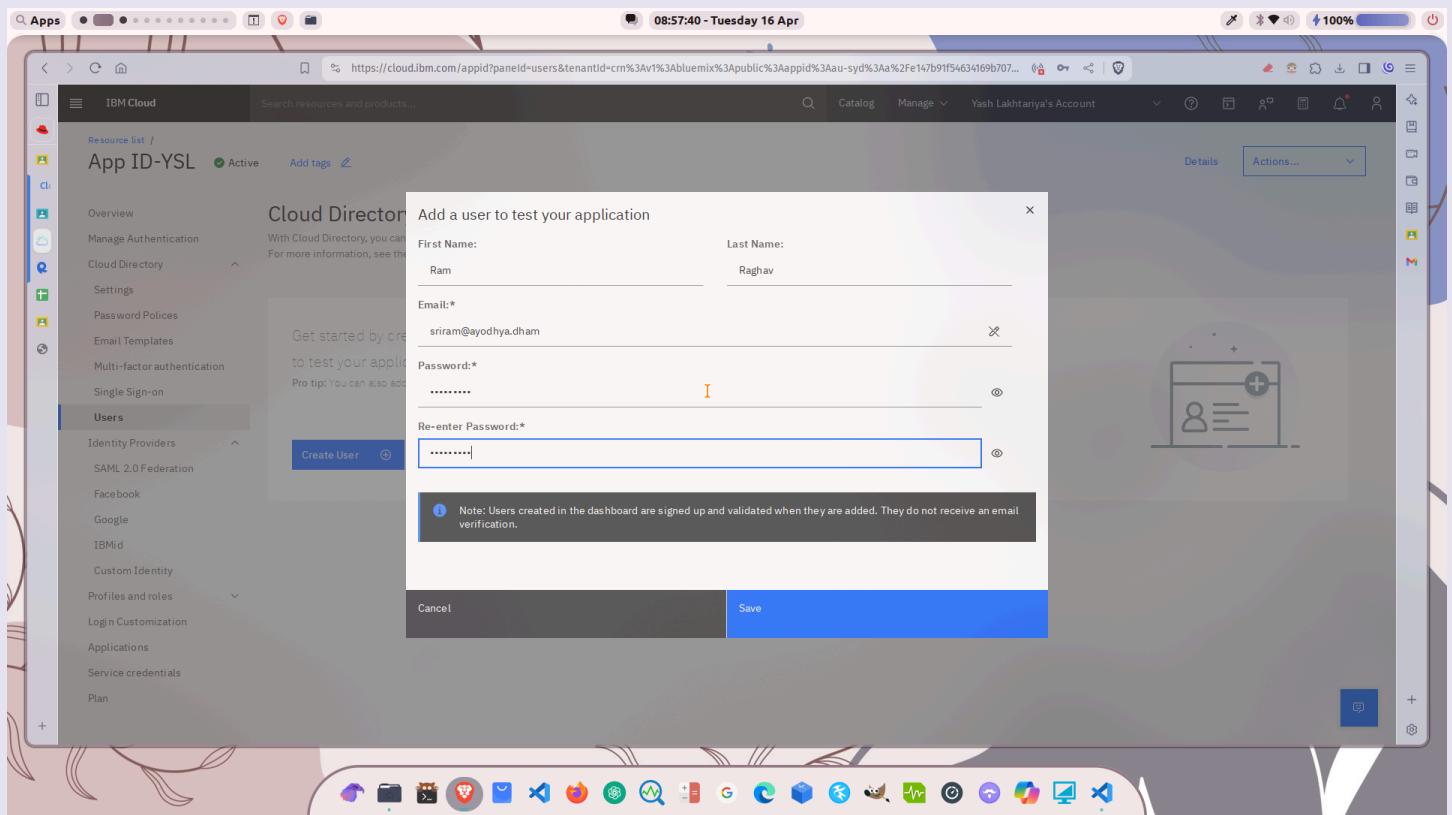
Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 61

EADC Practical 15

#### 4. Create a User in Cloud directory if required for later using so



Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## 5. Paste the credentials of application in the nodejs code

The screenshot shows a code editor interface with the following details:

- File Structure:** The project structure includes `node_modules`, `public`, `.env`, `app.js`, `app2.js`, `package-lock.json`, and `package.json`.
- Code Editor:** The main editor window displays `app.js` with the following code:

```
const express = require('express');
const session = require('express-session');
const passport = require('passport');
const WebAppStrategy = require('ibmcloud-appid').WebAppStrategy;
require('dotenv').config();

const app = express();
var port = 3000;
app.use(session({
  secret: '123456',
  resave: true,
  saveUninitialized: true
}));

app.get('/apid/login', passport.authenticate(WebAppStrategy.STRATEGY_NAME, {
  successRedirect: '/',
  forceLogin: true
}));

// Handle callback
app.get('/apid/callback', passport.authenticate(WebAppStrategy.STRATEGY_NAME, { keepSessionInfo: true }));
//app.use(passport.authenticate(WebAppStrategy.STRATEGY_NAME));
app.use('/api', (req, res, next) => {
  if (req.user) {
    next();
  } else {
    res.status(401).send("Unauthorized");
  }
}); // #39-45 app.use

app.get('/api/user', (req, res) => {
  //console.log(res);
  console.log(req.session[WebAppStrategy.AUTH_CONTEXT]);
});

app.listen(port, () => {
  console.log(`App listening on port ${port}!`);
  console.log(`Press Ctrl+C to stop.`);
});
```

- Bottom Bar:** Shows various icons for file operations, navigation, and developer tools.

Code :

```
const express = require('express');

const session = require('express-session');
const passport = require('passport');
const WebAppStrategy = require('ibmcloud-appid').WebAppStrategy;
require('dotenv').config();

const app = express();
var port = 3000;
app.use(session({
  secret: '123456',
  resave: true,
  saveUninitialized: true
}));

app.get('/apid/login', passport.authenticate(WebAppStrategy.STRATEGY_NAME, {
  successRedirect: '/',
  forceLogin: true
}));

// Handle callback
app.get('/apid/callback', passport.authenticate(WebAppStrategy.STRATEGY_NAME, { keepSessionInfo: true }));
//app.use(passport.authenticate(WebAppStrategy.STRATEGY_NAME));
app.use('/api', (req, res, next) => {
  if (req.user) {
    next();
  } else {
    res.status(401).send("Unauthorized");
  }
}); // #39-45 app.use

app.get('/api/user', (req, res) => {
  //console.log(res);
  console.log(req.session[WebAppStrategy.AUTH_CONTEXT]);
});

app.listen(port, () => {
  console.log(`App listening on port ${port}!`);
  console.log(`Press Ctrl+C to stop.`);
});
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

```
)}});  
  
app.use(passport.initialize());  
app.use(passport.session());  
  
  
passport.serializeUser((user, cb) => cb(null, user));  
passport.deserializeUser((user, cb) => cb(null, user));  
  
passport.use(new WebAppStrategy({  
    tenantId: process.env.tenantID,  
    clientId: process.env.clientID,  
    secret: process.env.secret,  
    oauthServerUrl: process.env.oAuthURL,  
    redirectUri: "http://localhost:3000/appid/callback"  
}));  
  
  
app.get('/appid/login',  
  passport.authenticate(WebAppStrategy.STRATEGY_NAME, {  
    successRedirect: '/',  
    forceLogin: true  
}));  
  
//  
// Handle callback  
app.get('/appid/callback',  
  passport.authenticate(WebAppStrategy.STRATEGY_NAME, { keepSessionInfo:  
true }));  
//app.use(passport.authenticate(WebAppStrategy.STRATEGY_NAME));  
app.use('/api', (req, res, next) => {  
  if (req.user) {  
    next();  
  } else {
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

```
    res.status(401).send("Unauthorized");
}

}) ;

app.get('/api/user', (req, res) => {
//console.log(res);
console.log(req.session[WebAppStrategy.AUTH_CONTEXT]);
res.json({
  user: {
    name: req.user.name
  }
});

}) ;

app.get("/appid/logout", function (req, res) {
  req.logout(function (err) {
    if (err) { console.log(err); }
    res.redirect("/");
  });
});

// Serve static resources

app.use(express.static('./public'));
app.listen(port);
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## 6. Now, add redirect URI to Manage Authentication > Authentication Settings > Add web redirect URIs

The screenshot shows the IBM Cloud App ID interface. The left sidebar lists 'Overview', 'Manage Authentication' (selected), 'Cloud Directory', 'Identity Providers', 'Profiles and roles', 'Login Customization', 'Applications', 'Service credentials', and 'Plan'. The main content area has tabs for 'Identity providers' and 'Authentication settings' (selected). Under 'Authentication settings', there is a section titled 'Add web redirect URLs' with the sub-instruction: 'Registering your redirect\_uri with App ID ensures that only authorized clients are allowed to participate in the authorization workflow. Make sure you register the redirect\_uri of applications that you trust. For more information about redirects, see the docs.' A text input field contains 'http://localhost:3000/appid/callback'. Below this, the 'Sign-in Expiration' section allows setting token lifetimes: 'Refresh token' (30 days, Disabled), 'Access token' (60 minutes), and 'Anonymous token' (30 days). The bottom navigation bar includes icons for various services like Watson, Blockchain, and Watson Studio.

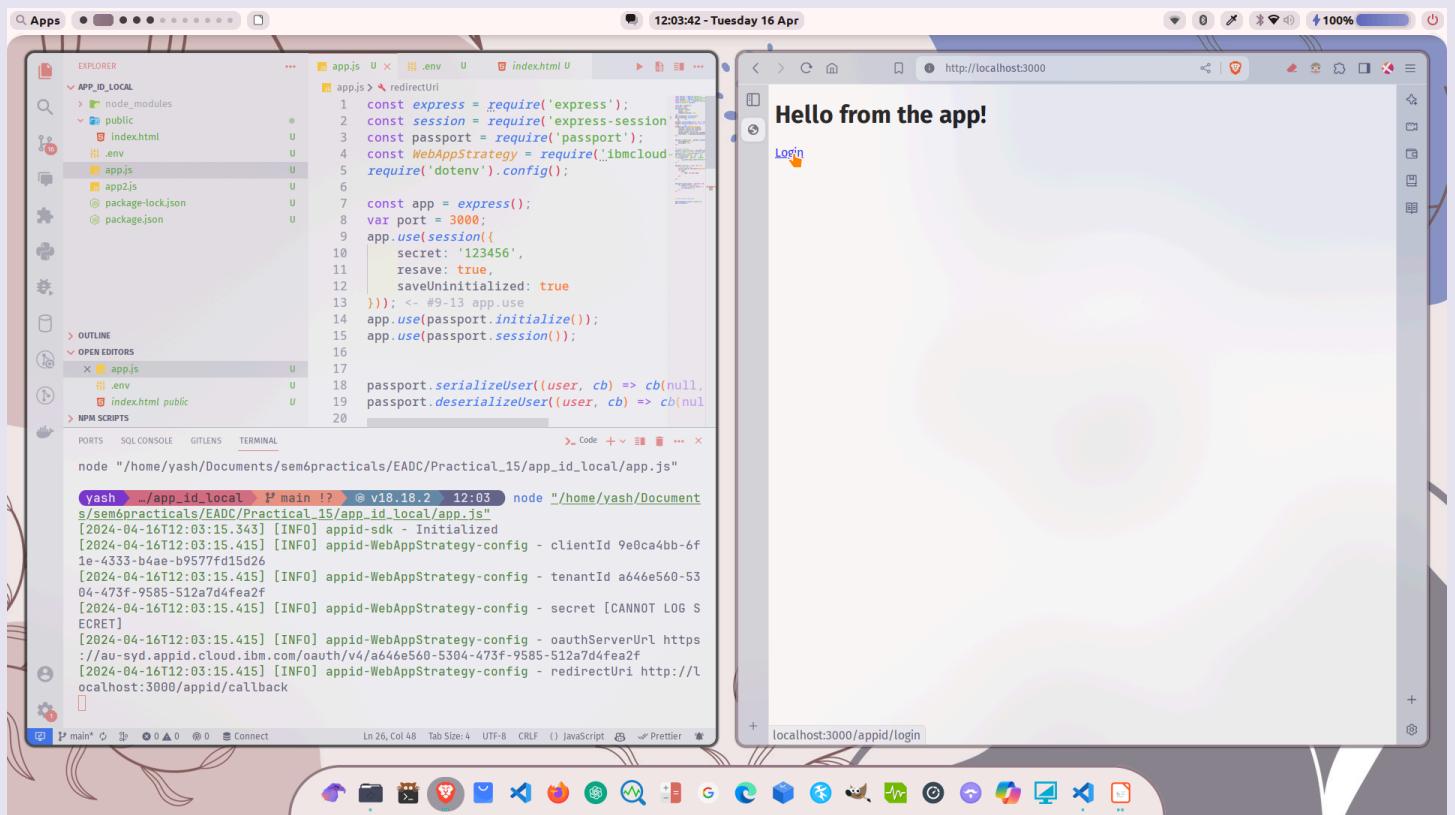
Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 61

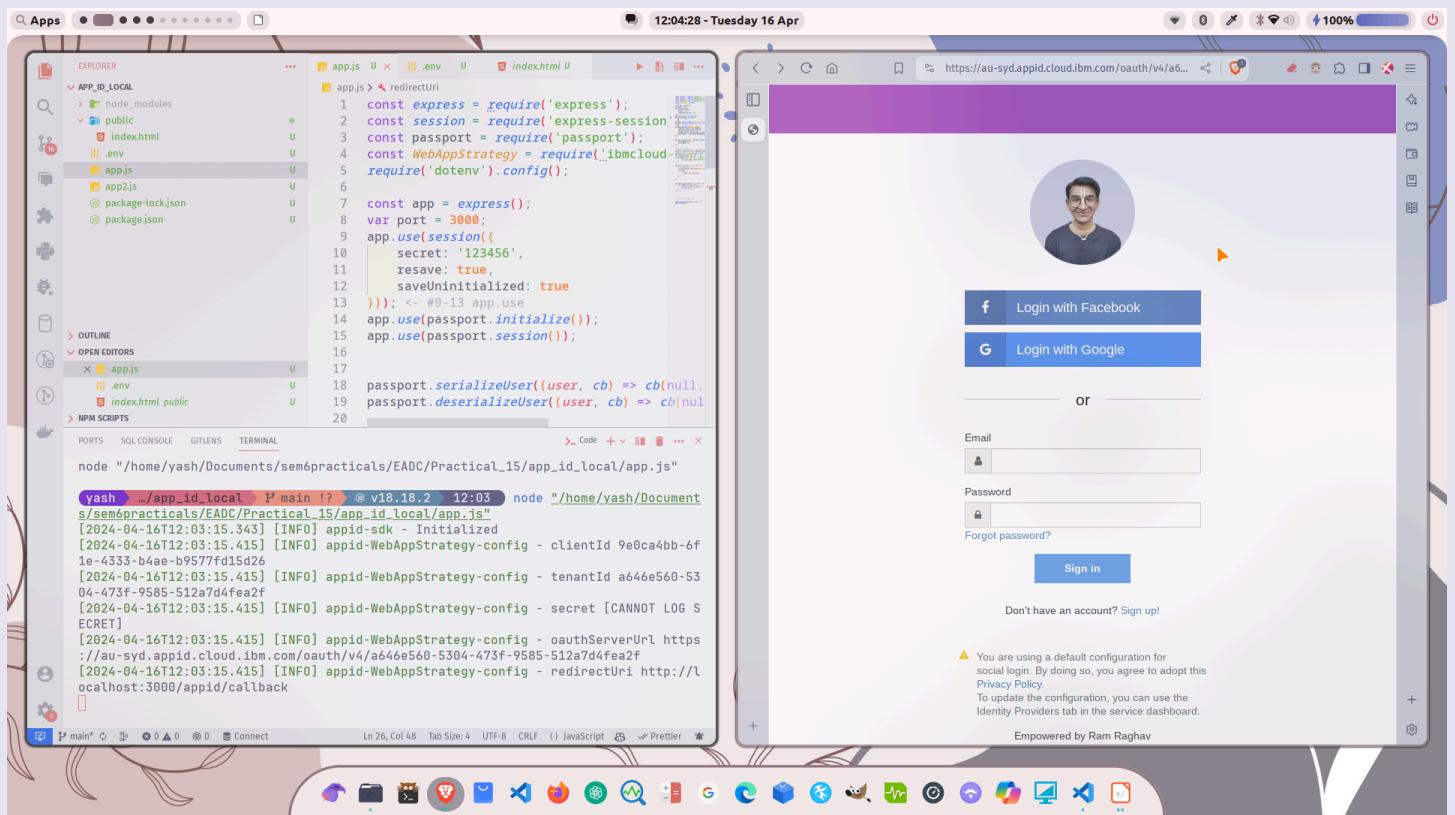
EADC Practical 15

## 7. Run app.js code and click login on the localhost page



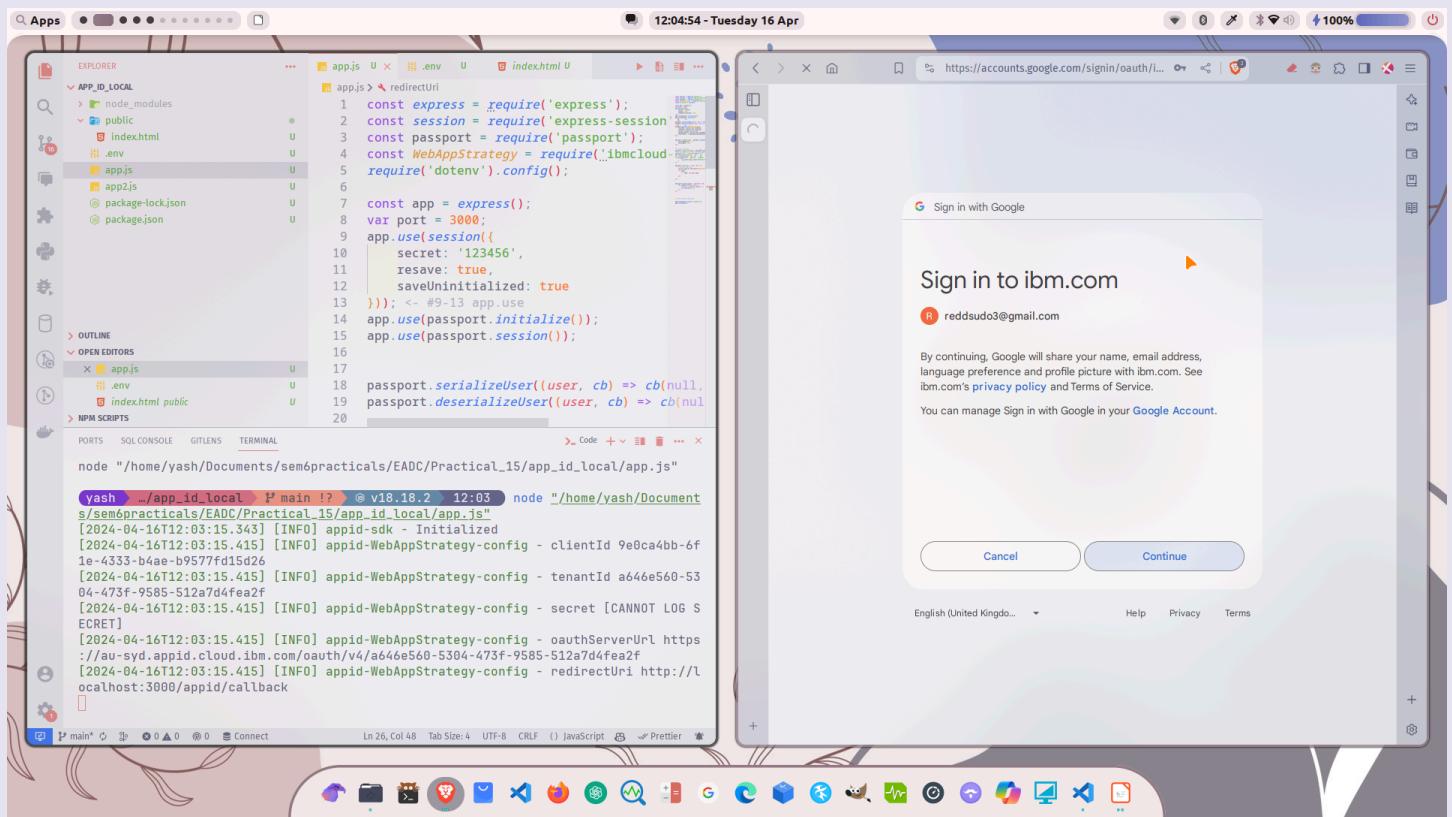
Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## 8. It will be redirected to login page of App ID service



Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## 9. Click login with google to check and login using the google account



Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## 10. Now, Logout button is visible and username is printed

The screenshot shows a Mac desktop environment. On the left is a code editor window for VS Code, displaying the file `app.js`. The code implements an Express application with session management and passport authentication. On the right is a web browser window showing the application's home page at `http://localhost:3000`. The page displays the message "Hello from the app!" and "Hello Redd Sudo". A "Logout" link is visible. The browser status bar indicates the time as 12:05:00 on Tuesday, 16 April.

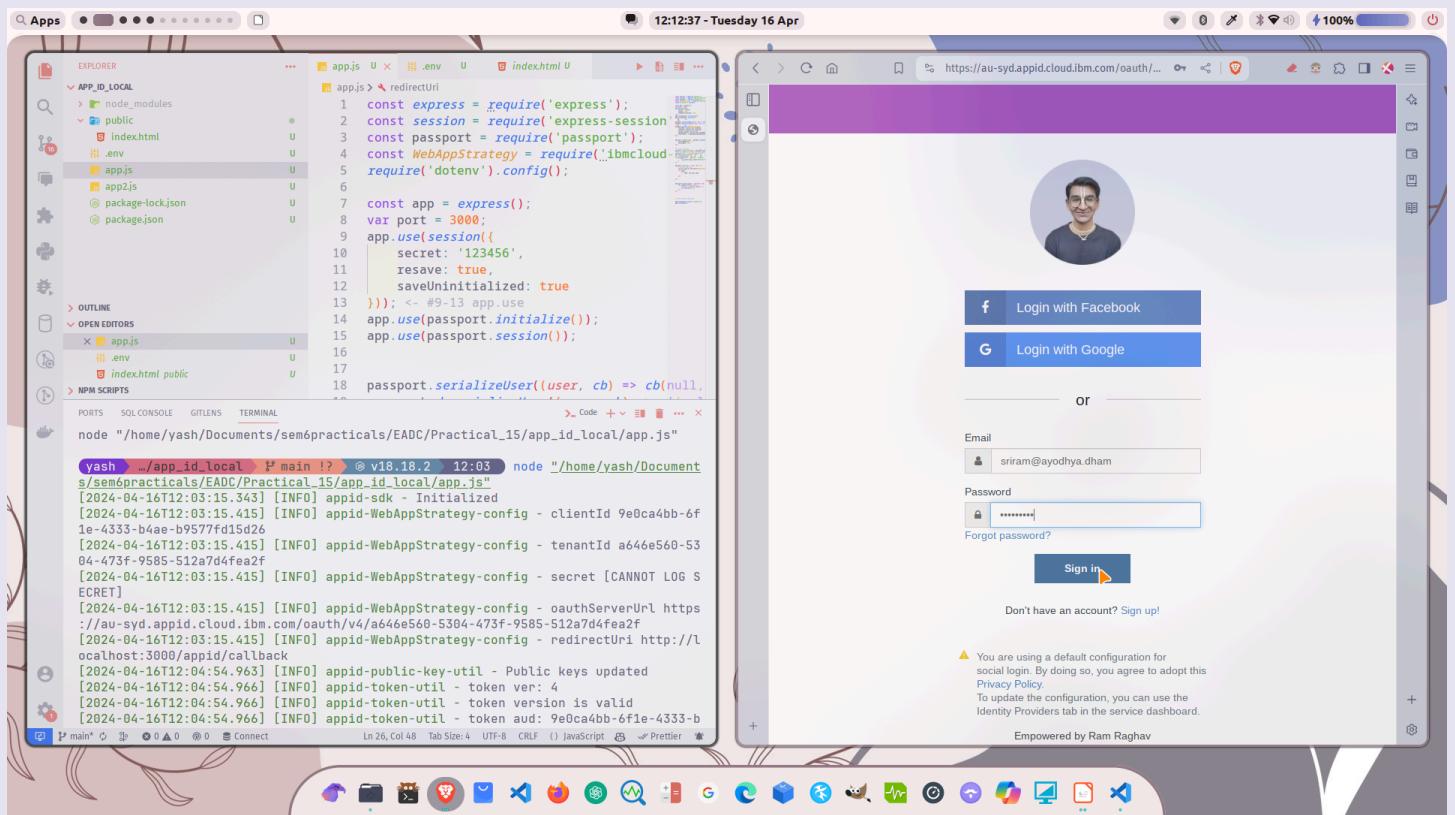
```
const express = require('express');
const session = require('express-session');
const passport = require('passport');
const WebAppStrategy = require('ibmcloud-passport-web');
const dotenv = require('dotenv').config();

const app = express();
var port = 3000;
app.use(session({
  secret: '123456',
  resave: true,
  saveUninitialized: true
})); // #9-13 app.use
app.use(passport.initialize());
app.use(passport.session());
```

```
exp: 1713252893,
tenant: 'a646e560-5304-473f-9585-512a7d4fea2f',
iat: 1713249293,
email: 'reddsudo@gmail.com',
name: 'Redd Sudo',
locale: 'en-US',
picture: 'https://lh3.googleusercontent.com/a/ACg8ocIjwTpknQAcDUEWov56kvYJ2zIa0zk4b16Kob_mV4qPZ4_qg=s96-c',
sub: 'c8083dbc-82aa-43b5-8543-57256720bca0',
given_name: 'Redd',
family_name: 'Sudo',
identities: [ [Object] ],
amr: [ 'google' ],
ver: 4
}
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

## 11. Try login using user created in App ID



Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

12. Now, the login is successful and copy the access token from console output to use afterwards in another authentication type (bearer auth) other than this one (user credentials)

The screenshot shows a macOS desktop environment. On the left is a code editor window for VS Code, displaying the file `app.js`. The code implements an Express application with session management and passport authentication. On the right is a web browser window showing the URL `http://localhost:3000`, which displays the message "Hello from the app!" and "Hello Ram Raghav". A "Logout" link is also visible. The system tray at the bottom shows various icons, and the status bar at the top indicates the date and time as "12:12:49 - Tuesday 16 Apr".

```
const express = require('express');
const session = require('express-session');
const passport = require('passport');
const WebAppStrategy = require('ibmcloud-appid').WebAppStrategy;
require('dotenv').config();

const app = express();
var port = 3000;
app.use(session({
    secret: '123456',
    resave: true,
    saveUninitialized: true
})); <- #9-13 app.use
app.use(passport.initialize());
app.use(passport.session());

passport.serializeUser((user, cb) => cb(null, user));
passport.deserializeUser((user, cb) => cb(null, user));

const redirectUri = process.env.REDIRECT_URI || 'http://localhost:3000/app/callback';

const appid_sdk = require('ibmcloud-appid-sdk');
const appid_webappstrategy_config = require('ibmcloud-appid-webappstrategy-config');

const appid_sdk_client_id = process.env.SDK_CLIENT_ID || '9e0ca4bb-6f1e-4333-b4ae-b9577fd15d26';
const appid_webappstrategy_config_tenant_id = process.env.WEBAPPSTRATEGY_CONFIG_TENANT_ID || 'a646e560-5304-473f-9585-512a7d4fea2f';

const appid_webappstrategy_config_options = {
    oauthServerUrl: `https://au-syd.appid.cloud.ibm.com/oauth/v4/a646e560-5304-473f-9585-512a7d4fea2f`,
    redirectUri: 'http://localhost:3000/app/callback'
};

const appid_public_key_util_options = {
    publicKeys: [
        {
            id: '54.966',
            key: '-----BEGIN PUBLIC KEY-----\nMIIBIjANBgcqhkjOPQ0NgwBAQHvJZGZDwQYJyWz\n-----END PUBLIC KEY-----'
        }
    ]
};

const appid_token_util_options = {
    tokenVersion: 4
};

const appid_token_util_audience = '9e0ca4bb-6f1e-4333-b4ae-b9577fd15d26';

const appid_token_util_options = {
    aud: appid_token_util_audience
};
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

13. In below code, paste the oAuthServerUrl and run the code

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows files like `APP_ID_LOCAL`, `node_modules`, `public`, `index.html`, `.env`, `app.js`, and `app2.js`.
- Editor:** The `app2.js` tab is active, displaying the following code:

```
const express = require('express');
const passport = require('passport');
const APIStrategy = require('ibmcloud-appid').APIStrategy;
require('dotenv').config();
const app = express();
app.use(passport.initialize());
passport.use(new APIStrategy({
  oauthServerUrl: process.env.oAuthURL,
}));
// Protect the whole app
app.use(passport.authenticate(APIStrategy.STRATEGY_NAME, {
  session: false
}));
// The /api/data API used to retrieve protected data
app.get('/api/data', (req, res) => {
  res.json([
    {
      data: 12345
    }
  ]);
}); // #17-21 app.get
app.listen(3000, () => {
  console.log('Listening on http://localhost:3000');
});

```

- Bottom Bar:** Shows various icons for file operations, navigation, and tools.
- Status Bar:** Shows "Ln 9, Col 1 (40 selected)" and "Tab Size: 4" and "UTF-8".

Code :

```
const express = require('express');
const passport = require('passport');
const APIStrategy = require('ibmcloud-appid').APIStrategy;
require('dotenv').config();
const app = express();
app.use(passport.initialize());
passport.use(new APIStrategy({
  oauthServerUrl: process.env.oAuthURL,
}));

// Protect the whole app
app.use(passport.authenticate(APIStrategy.STRATEGY_NAME, {
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 61  
EADC Practical 15

```
session: false
});

// The /api/data API used to retrieve protected data
app.get('/api/data', (req, res) => {
  res.json({
    data: 12345
  });
});

app.listen(3000, () => {
  console.log('Listening on http://localhost:3000');
});
```

14. Run the GET request to localhost:3000/api/data with Auth Type as Bearer token and paste the token there

