

ITA LAB 4

SAHIL BONDRE: U18CO021

AN INSTAGRAM CLIENT

Picture Search Engine

Create a simple CRUD app that interacts with a database. UI must be very clean and simple and it must be responsive.

Account is necessary to perform the Specified searches. Apply authentication and authorization mechanism (Only authorized user can be able to perform a CRUD operation on a document).

Use your favorite backend language which consists of a simple Search Engine to search the Pictures.

Different Search options are:

- A client can search the Pictures with the hashtag as a keyword for search.
- A client can provide the Account name as keyword for search.

The Pictures should be listed with the Profile of the person who has uploaded the pictures (latest First).

Download option should also be available for the Particular selected Picture.

An advance option should be available to filter the Pictures through Account, dates of creation, other hashtags or some keywords.

A resultant list must be in the pagination form at server and client side.

Implement an analytics like:

- How many times a particular HASHTAGS is searched.
- How many times each Picture is downloaded.

Backend Infrastructure

The backend is deployed as a set of microservices. I am using a combination of microservice and serverless applications:

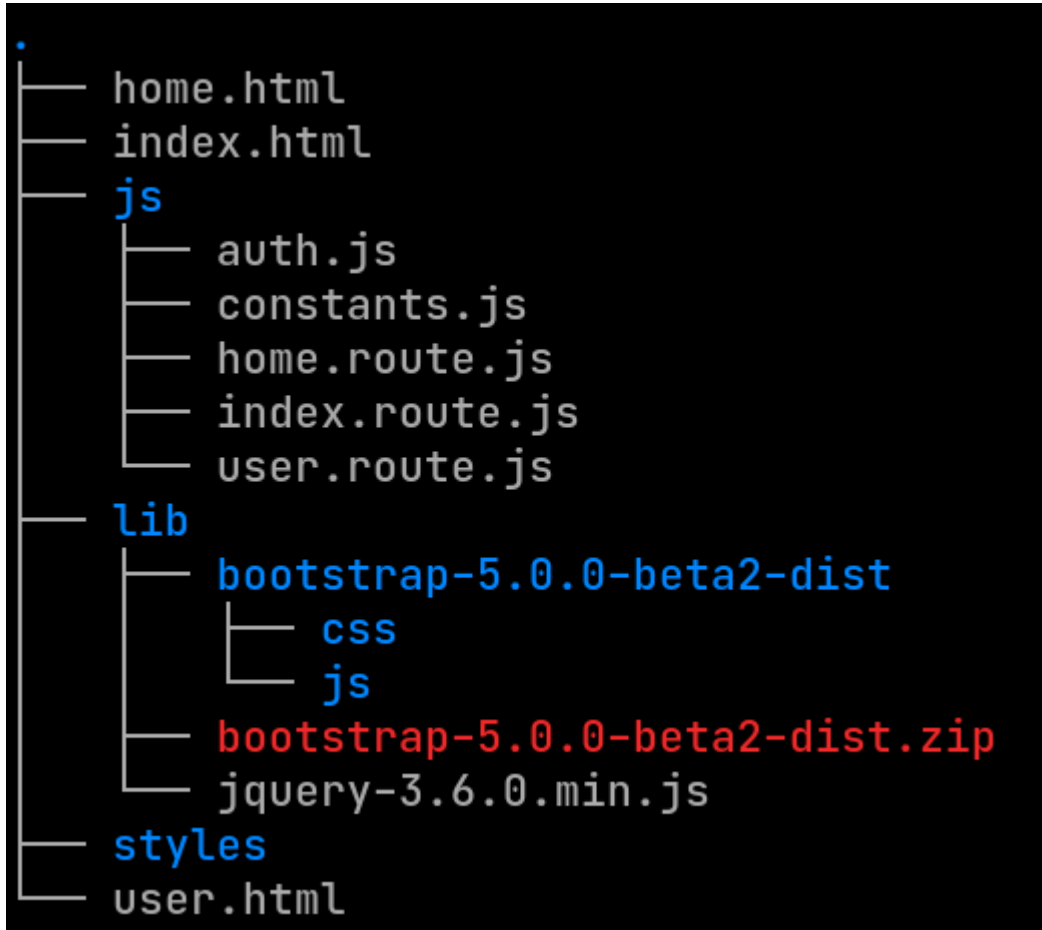
User, Authentication and Authorisation: Docker + NodeJS + Express + MongoDB services

Image and Analytics: Cloud Firestore

Image file storage: GCP Storage Bucket

Frontend Code

Directory Structure



home.html

```
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <link
      rel="stylesheet"
      href="lib/bootstrap-5.0.0-beta2-dist/css/bootstrap.min.css"
    />
    <script
src="lib/bootstrap-5.0.0-beta2-dist/js/bootstrap.bundle.min.js"></script
```

```

>
  <title>Home Page</title>
</head>
<body>
  <div
    class="modal fade"
    id="exampleModal"
    tabindex="-1"
    aria-labelledby="exampleModalLabel"
    aria-hidden="true"
  >
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-title" id="exampleModalLabel">Modal
title</h5>
          <button
            type="button"
            class="btn-close"
            data-bs-dismiss="modal"
            aria-label="Close"
          ></button>
        </div>
        <div id="modal-body" class="modal-body">...</div>
        <div class="modal-footer">
          <button
            type="button"
            class="btn btn-secondary"
            data-bs-dismiss="modal"
          >
            Close
          </button>
        </div>
      </div>
    </div>
  </div>
</div>
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">InstaSearch</a>
    <button
      class="navbar-toggler"
      type="button"
      data-bs-toggle="collapse"
      data-bs-target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent"
      aria-expanded="false"
    >

```

```

        aria-label="Toggle navigation"
    >
        <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse"
id="navbarSupportedContent">
        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
            <li class="nav-item">
                <a class="nav-link" aria-current="page"
href="/home">Home</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="/user">Profile</a>
            </li>
        </ul>

        <button
            class="btn btn-outline-success"
            id="logout-button"
            type="submit"
        >
            Logout
        </button>
    </div>
</div>
</nav>
<div class="container">
    <div class="row align-items-center mt-5">
        <div class="col">
            <div class="mb-3">
                <label for="tag-input" class="form-label">Search by
Hashtag</label>
                <input type="text" class="form-control" id="tag-input" />
            </div>
            <button class="btn btn-primary"
id="tag-search">Search</button>
        </div>
        <div class="col">
            <div class="mb-3">
                <label for="user-input" class="form-label">Search by
User</label>
                <input type="text" class="form-control" id="user-input" />
            </div>
            <button class="btn btn-primary"
id="user-search">Search</button>
        </div>
    </div>

```

```

</div>

<div class="mt-2">
  <span id="search-badge" hidden="true" class="badge bg-success"
    >Search Count: 45</span>
  </div>

  <div class="d-flex flex-wrap p-2" id="image-grid"></div>
</div>
<script
src="https://www.gstatic.com/firebasejs/8.3.0/firebase-app.js"></script>

  <!-- TODO: Add SDKs for Firebase products that you want to use
  https://firebase.google.com/docs/web/setup#available-libraries -->
  <script
src="https://www.gstatic.com/firebasejs/8.3.0/firebase-storage.js"></scr
ipt>
  <script
src="https://www.gstatic.com/firebasejs/8.3.0/firebase-firestore.js"></s
cript>
  <script>
    // TODO: Add your web app's Firebase configuration
    // Initialize Firebase
    firebase.initializeApp(firebaseConfig);
  </script>
  <script src="lib/jquery-3.6.0.min.js"></script>
  <script src="js/constants.js"></script>
  <script src="js/auth.js"></script>
  <script src="js/home.route.js"></script>
</body>
</html>

```

home.route.js

```

$("#logout-button").on("click", () => {
  localStorage.removeItem("token");
  window.location = "/";
});

const getUrlParams = (url) => {
  const params = {};
  (url + "?")
    .split("?")[1]

```

```

        .split("&")
        .forEach(function (pair) {
            pair = (pair + "=").split("=").map(decodeURIComponent);
            if (pair[0].length) {
                params[pair[0]] = pair[1];
            }
        });
    return params;
};

const storage = firebase.storage();
const db = firebase.firestore();

(async () => {
    const makeCard = (url, username, id, caption, filename) => {
        console.log(filename);
        return $(`<div class="card m-5" style="width: 18rem">
            
            <div class="card-body">
                <a class="card-title" href="/user?id=${id}">${username}</a>
                <p class="card-text">${caption}
            </p>
            </div>
        </div>`);
    };

    $("#tag-search").on("click", () => {
        const tag = $("#tag-input").val();
        db.collection("images")
            .where("tags", "array-contains", tag)
            .get()
            .then((snapshot) => {
                console.log(snapshot);
                snapshot.docs
                    .map((e) => e.data())
                    .sort((a, b) => b.time - a.time)
                    .forEach((data) => {
                        console.log(data);
                        $("#image-grid").append(
                            makeCard(
                                data.url,
                                data.username,

```

```

        data.user,
        data.caption,
        data.fileName
    )
    );
    });
});

db.collection("meta")
  .doc("tags")
  .get()
  .then((doc) => {
    const data = doc.data();
    console.log("data", data);
    document.getElementById("search-badge").hidden = false;
    if (data[tag] === undefined) {
      // first time search
      let obj = {};
      obj[tag] = 1;
      db.collection("meta").doc("tags").set(obj, { merge: true });
      document.getElementById("search-badge").innerHTML =
        "Search Count: " + 1;
    } else {
      const val = data[tag];
      let obj = {};
      obj[tag] = Number(val) + 1;
      db.collection("meta").doc("tags").set(obj, { merge: true });
      document.getElementById("search-badge").innerHTML =
        "Search Count: " + (Number(val) + 1);
    }
  });
});

$("#user-search").on("click", async () => {
  const name = $("#user-input").val();
  const token = localStorage.getItem("token");

  const res = await fetch(`${BASE_URL}/users?name=${name}`, {
    method: "GET",
    headers: {
      ...headers,
      Authorization: `Bearer ${token}`,
    },
  });
});
const json = await res.json();
const users = json.results;

```

```

if (users === undefined || users.length === 0) {
  $("#modal-body").html("No user found");
  $("#exampleModal").modal("show");
  return;
}
const userId = users[0].id;
console.log("userId", userId);
db.collection("images")
  .where("user", "=", userId)
  .get()
  .then((snapshot) => {
    console.log(snapshot.docs);
    snapshot.docs
      .map((e) => e.data())
      .sort((a, b) => b.time - a.time)
      .forEach((data) => {
        console.log(data);
        $("#image-grid").append(
          makeCard(
            data.url,
            data.username,
            data.user,
            data.caption,
            data.fileName
          )
        );
      });
  });

db.collection("meta")
  .doc("tags")
  .get()
  .then((doc) => {
    const data = doc.data();
    console.log("data", data);
    document.getElementById("search-badge").hidden = false;
    if (data[name] === undefined) {
      // first time search
      let obj = {};
      obj[name] = 1;
      db.collection("meta").doc("tags").set(obj, { merge: true });
      document.getElementById("search-badge").innerHTML =
        "Search Count: " + 1;
    } else {
      const val = data[name];
      let obj = {};

```



```

        obj[name] = Number(val) + 1;
        db.collection("meta").doc("tags").set(obj, { merge: true });
        document.getElementById("search-badge").innerHTML =
            "Search Count: " + (Number(val) + 1);
    }
    });
    });
    })().catch((err) => {
        console.error(err);
    });

```

index.html

```

<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <script
src="lib/bootstrap-5.0.0-beta2-dist/js/bootstrap.bundle.min.js"></script>
  >
    <script src="lib/jquery-3.6.0.min.js"></script>
    <link
      rel="stylesheet"
      href="lib/bootstrap-5.0.0-beta2-dist/css/bootstrap.min.css"
    />
    <title>InstaSearch</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
      <div class="container-fluid">
        <a class="navbar-brand" href="/">InstaSearch</a>
        <button
          class="navbar-toggler"
          type="button"
          data-bs-toggle="collapse"
          data-bs-target="#navbarSupportedContent"
          aria-controls="navbarSupportedContent"
          aria-expanded="false"
          aria-label="Toggle navigation"
        >
          <span class="navbar-toggler-icon"></span>
        </button>

```

```

        <div class="collapse navbar-collapse"
id="navbarSupportedContent">
            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                <li class="nav-item">
                    <a class="nav-link active" aria-current="page"
href="/">Home</a>
                </li>
            </ul>
        </div>
    </div>
</nav>
<div
    class="modal fade"
    id="exampleModal"
    tabindex="-1"
    aria-labelledby="exampleModallLabel"
    aria-hidden="true"
>
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModallLabel">Modal
title</h5>
                <button
                    type="button"
                    class="btn-close"
                    data-bs-dismiss="modal"
                    aria-label="Close"
                ></button>
            </div>
            <div id="modal-body" class="modal-body">...</div>
            <div class="modal-footer">
                <button
                    type="button"
                    class="btn btn-secondary"
                    data-bs-dismiss="modal"
                >
                    Close
                </button>
            </div>
        </div>
    </div>
</div>
<div class="container mt-5">
    <div class="row align-items-center">
        <div class="col">

```

```

    <h1 class="display-3">Welcome to InstaSearch</h1>
    <h3>Sign In or Register to Continue</h3>
</div>
<div class="col px-5">
    <h1>Register</h1>
    <div class="mb-3">
        <label for="register-username"
class="form-label">Username</label>
        <input
            type="text"
            name=""
            class="form-control"
            id="register-username"
        />
    </div>
    <div class="mb-3">
        <label for="exampleInputEmail1"
class="form-label">Email</label>
        <input
            type="email"
            class="form-control"
            aria-describedby="emailHelp"
            id="register-email"
        />
        <div id="emailHelp" class="form-text">
            We'll never share your email with anyone else.
        </div>
    </div>
    <div class="mb-3">
        <label for="exampleInputPassword1" class="form-label"
        >Password</label>
        >
        <input
            type="password"
            class="form-control"
            id="register-password"
        />
    </div>
    <button type="submit" id="register-button" class="btn
btn-primary">
        Register
    </button>
</div>
<div class="col">
    <h1>Login</h1>
    <div class="mb-3">

```

```

        <label for="exampleInputEmail1"
class="form-label">Email</label>
        <input
            type="email"
            class="form-control"
            aria-describedby="emailHelp"
            id="login-email"
        />
        <div id="emailHelp" class="form-text">
            We'll never share your email with anyone else.
        </div>
    </div>
    <div class="mb-3">
        <label for="exampleInputPassword1" class="form-label"
        >Password</label>
        >
        <input type="password" class="form-control"
id="login-password" />
    </div>
    <button type="submit" id="login-button" class="btn
btn-primary">
        Login
    </button>
    </div>
</div>
</div>
<script src="js/constants.js"></script>
<script src="js/auth.js"></script>
<script src="js/index.route.js"></script>
</body>
</html>

```

index.route.js

```

$("#register-button").on("click", async () => {
    const username = $("#register-username").val();
    const email = $("#register-email").val();
    const password = $("#register-password").val();

    console.log(username, email, password);
    if (username === undefined || username.length <= 3) {
        $("#modal-body").html("Invalid Username");
        $("#exampleModal").modal("show");
        return;
    }

```

```

}
if (!validateEmail(email)) {
  $("#modal-body").html("Invalid Email");
  $("#exampleModal").modal("show");
  return;
}
if (password === undefined || password.length < 8) {
  $("#modal-body").html("Password must be atleast 8 charecters long");
  $("#exampleModal").modal("show");
  return;
}
try {
  const res = await fetch(`${BASE_URL}/auth/register`, {
    method: "POST",
    headers,
    body: JSON.stringify({
      name: username,
      password: password,
      email: email,
    }),
  });

  if (!res.ok) {
    throw new Error("Email exists");
  }

  const json = await res.json();
  console.log(json);
  $("#modal-body").html("New user created! Login to continue");
  $("#exampleModal").modal("show");
  $("#input").val("");
} catch (err) {
  console.log(err.toString());
  $("#modal-body").html("Email already exists");
  $("#exampleModal").modal("show");
  return;
}

console.log("Reached!");
});

$("#login-button").on("click", async () => {
  const email = $("#login-email").val();
  const password = $("#login-password").val();

  console.log(email, password);

```

```

if (!validateEmail(email)) {
  $("#modal-body").html("Invalid Email");
  $("#exampleModal").modal("show");
  return;
}
if (password === undefined || password.length < 8) {
  $("#modal-body").html("Password must be atleast 8 charecters long");
  $("#exampleModal").modal("show");
  return;
}
try {
  const res = await fetch(`${BASE_URL}/auth/login`, {
    method: "POST",
    headers,
    body: JSON.stringify({
      password,
      email,
    }),
  });

  if (!res.ok) {
    throw new Error("Invalid Login");
  }

  const json = await res.json();
  console.log(json);
  localStorage.setItem("token", json.tokens.access.token);
  localStorage.setItem("email", email);
  localStorage.setItem("id", json.user.id);
  localStorage.setItem("username", json.user.name);
  $("#modal-body").html("Login Successful!");
  $("#exampleModal").modal("show");
  $("#input").val("");
  window.location = "/home";
} catch (err) {
  console.log(err.toString());
  $("#modal-body").html("Invalid Credentials");
  $("#exampleModal").modal("show");
  return;
}

console.log("Reached!");
});

function validateEmail(email) {
  const re =

```

```

/^(([<>()[]\.,;:\s@"]+(\.[<>()[]\.,;:\s@"]+)*)|("[.+"))@((\[0-9]{1,3}\. [0-9]{1,3}\. [0-9]{1,3}\. [0-9]{1,3}\)|((([a-zA-Z0-9]+\.)+[a-zA-Z]{2,}))$)/;
    return re.test(String(email).toLowerCase());
}

```

user.html

```

<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <link
      rel="stylesheet"
      href="lib/bootstrap-5.0.0-beta2-dist/css/bootstrap.min.css"
    />
    <script
src="lib/bootstrap-5.0.0-beta2-dist/js/bootstrap.bundle.min.js"></script
  >
    <title>User Page</title>
  </head>
  <body>
    <div
      class="modal fade"
      id="exampleModal"
      tabindex="-1"
      aria-labelledby="exampleModalLabel"
      aria-hidden="true"
    >
      <div class="modal-dialog">
        <div class="modal-content">
          <div class="modal-header">
            <h5 class="modal-title" id="exampleModalLabel">Modal
title</h5>
            <button
              type="button"
              class="btn-close"
              data-bs-dismiss="modal"
              aria-label="Close"
            ></button>
          </div>
          <div id="modal-body" class="modal-body">...</div>

```

```

        <div class="modal-footer">
            <button
                type="button"
                class="btn btn-secondary"
                data-bs-dismiss="modal"
            >
                Close
            </button>
        </div>
    </div>
</div>
<div>
<nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container-fluid">
        <a class="navbar-brand" href="#">InstaSearch</a>
        <button
            class="navbar-toggler"
            type="button"
            data-bs-toggle="collapse"
            data-bs-target="#navbarSupportedContent"
            aria-controls="navbarSupportedContent"
            aria-expanded="false"
            aria-label="Toggle navigation"
        >
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse"
id="navbarSupportedContent">
            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                <li class="nav-item">
                    <a class="nav-link" aria-current="page"
href="/home">Home</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="/user">Profile</a>
                </li>
            </ul>

            <button
                class="btn btn-outline-success"
                id="logout-button"
                type="submit"
            >
                Logout
            </button>
        </div>
    </nav>

```



```

    </div>
  </nav>
  <div class="container">
    <div class="row align-items-center mt-5">
      <div class="col">
        <h1 id="username-text">Username</h1>
        <h3 id="email-text">email@example.com</h3>
      </div>
      <div class="col"></div>
      <div class="col user-only" id="upload-form">
        <input
          id="upload"
          type="file"
          accept="image/*"
          class="form-control border-1 my-2"
        />
        <div class="form-floating my-2">
          <textarea
            class="form-control"
            placeholder="Leave a caption here"
            id="floatingTextarea2"
            style="height: 100px"
          ></textarea>
          <label for="floatingTextarea2">Caption</label>
        </div>
        <button id="add-button" class="btn btn-primary">Add
Image</button>
      </div>
    </div>

    <div class="d-flex flex-wrap p-2" id="image-grid">
    </div>
  </div>
  <!-- The core Firebase JS SDK is always required and must be listed
first -->
  <script
src="https://www.gstatic.com/firebasejs/8.3.0/firebase-app.js"></script>

  <!-- TODO: Add SDKs for Firebase products that you want to use
https://firebase.google.com/docs/web/setup#available-libraries -->
  <script
src="https://www.gstatic.com/firebasejs/8.3.0/firebase-storage.js"></scr
ipt>
  <script
src="https://www.gstatic.com/firebasejs/8.3.0/firebase-firestore.js"></s
cript>

```

```

<script>
  // TODO: Your web app's Firebase configuration
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
<script src="lib/jquery-3.6.0.min.js"></script>
<script src="js/constants.js"></script>
<script src="js/auth.js"></script>
<script src="js/user.route.js"></script>
</body>
</html>

```

user.route.js

```

const storage = firebase.storage();
const db = firebase.firestore();
console.log(storage);

async function deleteImage(fileName) {
  const res = await db
    .collection("images")
    .where("fileName", "==", fileName)
    .get();
  const id = res.docs[0].id;

  await db.collection("images").doc(id).delete();
  const imgRef = storage.ref().child(fileName);
  await imgRef.delete();

  $("#modal-body").html("Deleted");
  $("#exampleModal").modal("show");
}

function openInNewTab(href) {
  Object.assign(document.createElement("a"), {
    target: "_blank",
    href: href,
  }).click();
}

async function incrementDownloadCount(fileName) {
  const res = await db
    .collection("images")
    .where("fileName", "==", fileName)

```

```

        .get();
const id = res.docs[0].id;
const count = res.docs[0].data().count + 1;
const url = res.docs[0].data().url;
console.log(res.docs[0].data().count);
console.log(count);

await db.collection("images").doc(id).set({ count }, { merge: true });

openInNewTab(url);
}

(async () => {
  $("#logout-button").on("click", () => {
    localStorage.removeItem("token");
    window.location = "/";
  });

  const makeCard = (url, username, id, caption, filename, count) => {
    console.log(filename);
    return `(
      <div class="card m-5" style="width: 18rem">
        
        <div class="card-body">
          <a class="card-title" href="/user?id=${id}">${username}</a>
          <p class="card-text">${caption}
          </p>
          <div class="card-footer text-muted">
            ${count} downloads
          </div>
          <button class="btn btn-primary"
onclick="incrementDownloadCount('${filename}')">Download</button>
          <button class="btn btn-danger user-only"
onclick="deleteImage('${filename}')">Delete</button>

        </div>
      </div>`);
  };

  const username = localStorage.getItem("username");
  const email = localStorage.getItem("email");
  const id = localStorage.getItem("id");
  const token = localStorage.getItem("token");

```

```

const getUrlParams = (url) => {
  const params = {};
  (url + "?")
    .split("?")[1]
    .split("&")
    .forEach(function (pair) {
      pair = (pair + "=").split("=").map(decodeURIComponent);
      if (pair[0].length) {
        params[pair[0]] = pair[1];
      }
    });
  return params;
};

const params = getUrlParams(window.location.href);
console.log(params);
if (params.id === undefined || params.id === id) {
  // self
  console.log("SELF");
  $("#username-text").html(username);
  $("#email-text").html(email);
  $(".user-only").show();
} else {
  console.log(token);
  // other user
  console.log("OTHER");
  const res = await fetch(`${BASE_URL}/users/${params.id}`, {
    method: "GET",
    headers: {
      ...headers,
      Authorization: `Bearer ${token}`,
    },
  });
  if (!res.ok) {
    window.location = "/user";
  }

  const json = await res.json();
  const username = json.name;
  const email = json.email;
  $("#username-text").html(username);
  $("#email-text").html(email);
  $(".user-only").hide();
  console.log(json);
}

```

```

}

$("#add-button").on("click", async () => {
  const file = document.getElementById("upload").files[0];
  const caption = $("#textarea").val();
  let tags = caption.match(/#[a-z]+/gi);
  if (tags === null) tags = [];
  const id = localStorage.getItem("id");
  const time = Number(new Date()) + "";
  const fileName = `images/${id + time}`;

  if (file === undefined) {
    $("#modal-body").html("Select an image first");
    $("#exampleModal").modal("show");
    return;
  }

  if (caption.length === 0) {
    $("#modal-body").html("Caption cannot be empty");
    $("#exampleModal").modal("show");
    return;
  }

  const imgRef = storage.ref().child(fileName);

  await imgRef.put(file);
  const url = await imgRef.getDownloadURL();
  console.log(url);

  await db.collection("images").add({
    user: id,
    url,
    count: 0,
    caption,
    tags,
    time: Number(Date.now()),
    username,
    fileName,
  });

  $("#modal-body").html("Image uploaded!");
  $("#exampleModal").modal("show");
  $("#input").val("");
});

let userId = params.id || localStorage.getItem("id");

```

```

console.log(userId);
db.collection("images")
  .where("user", "==", userId)
  .get()
  .then((snapshot) => {
    console.log(snapshot);
    snapshot.docs
      .map((e) => e.data())
      .sort((a, b) => b.time - a.time)
      .forEach((data) => {
        console.log(data);
        $("#image-grid").append(
          makeCard(
            data.url,
            data.username,
            data.user,
            data.caption,
            data.fileName,
            data.count
          )
        );
      });
    if (params.id !== undefined && params.id !== id) {
      $(".user-only").hide();
    }
  });
})();
}).catch((err) => {
  console.error(err);
});

```

auth.js

```

if (window.location.pathname !== "/" && localStorage.getItem("token")
=== null) {
  window.location = "/";
}

if(window.location.pathname === "/" && localStorage.getItem("token") !==
null) {
  window.location = "/home"
}

```

constants.js

```
const BASE_URL = "http://localhost:3000/v1";
const headers = {
  Accept: "application/json",
  "Content-Type": "application/json",
};
```

Client Side Libraries Used:

- JQuery
- Bootstrap

Screenshots:

Login/Register Page:

InstaSearch Home

Welcome to InstaSearch

Sign In or Register to Continue

Register

Username

Email

We'll never share your email with anyone else.

Password

Register

Login

Email

We'll never share your email with anyone else.

Password

Login

User Home Page with search options:

InstaSearch

[Home](#)[Profile](#)[Logout](#)

Search by Hashtag

Search

Search by User

Search

User profile page:

InstaSearch

Home

Profile

Logout

Sahil

admin@example.com

Choose File

No file chosen

Caption

Add Image

0

0

0

No rides found!

Connect with apps to fetch rides

Connect now

Sahil

#cycle #blast "this is not a hashtag"

6 downloads

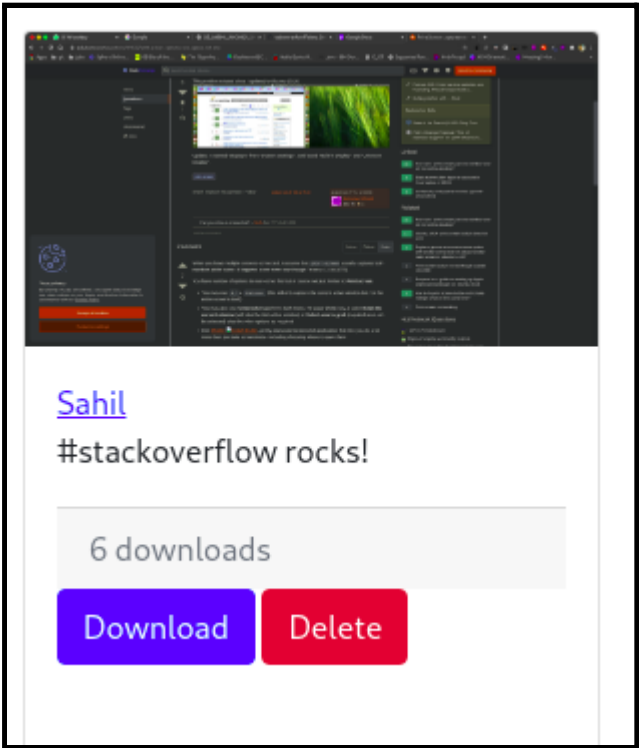
Download

Delete

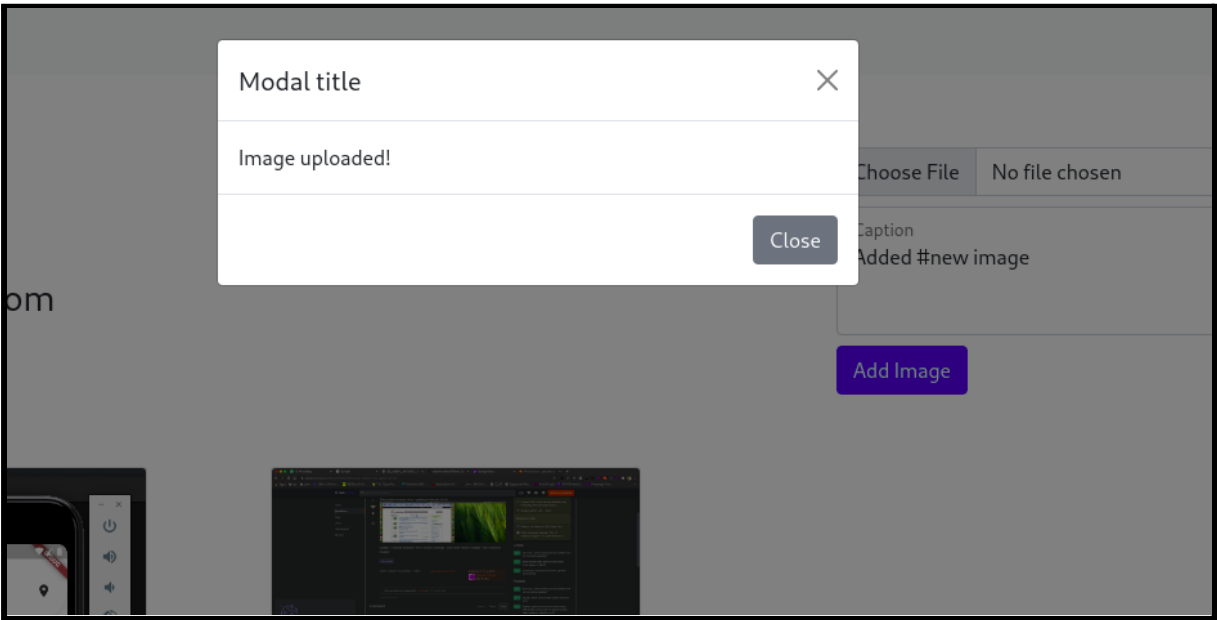
Sahil

#stackoverflow rocks!

User post with option to download, delete and download analytics:



User Image upload modal:



Example of search by hashtag:

Search by Hashtag

#cycle

Search

Search Count: 7

Search by User

Search

0

0

No rides found!

Connect with apps to fetch rides.

Connect now

Sahil

#cycle #blast "this is not a hashtag"

Hashtag search analytics:

Search by Hashtag

Example of search by username:

Search by Hashtag

#cycle

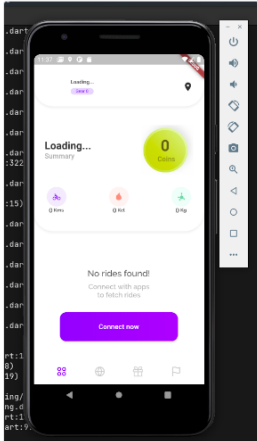
Search

Search Count: 6


Search by User

godcrampy


Search



[Sahil](#)
#cycle #blast "this is not a hashtag"



[godcrampy](#)
this is #ACM




[godcrampy](#)
This is #turing

Viewing other users profile

[InstaSearch](#) [Home](#) [Profile](#) [Logout](#)

godcrampy


sahilbondre@gmail.com



[godcrampy](#)
this is #ACM

9 downloads

Download




[godcrampy](#)
This is #turing

80 downloads

Download

Example of other users post with download option and download analytics:



[godcrampy](#)
This is #turing

80 downloads

Download

User Authentication Service running locally:

```
→ server git:(master) X yarn start
yarn run v1.21.1

$ pm2 start ecosystem.config.json --no-daemon

pm2 launched in no-daemon mode (you can add DEBUG="*" env variable to get more messages)
2021-03-26T22:09:59: PM2 log: Launching in no daemon mode
2021-03-26T22:09:59: PM2 log: [PM2][WARN] Applications app not running, starting...
2021-03-26T22:09:59: PM2 log: App [app:0] starting in -cluster mode-
2021-03-26T22:09:59: PM2 log: App [app:0] online
2021-03-26T22:09:59: PM2 log: [PM2] App [app] launched (1 instances)
2021-03-26T22:09:59: PM2 log:
```

id	name	namespace	version	mode	pid	uptime	↺	status	c
pu	mem	user	watching						
0	app	default	1.6.0	cluster	134033	0s	0	online	0
%	33.8mb	sahil	disabled						

```
2021-03-26T22:09:59: PM2 log: [--no-daemon] Continue to stream logs
2021-03-26T22:09:59: PM2 log: [--no-daemon] Exit on target PM2 exit pid=134022
22:10:00 0|app | 2021-03-26T22:10:00: (node:134033) Warning: Accessing non-existent property
'MongoError' of module exports inside circular dependency
22:10:00 0|app | (Use `node --trace-warnings ...` to show where the warning was created)
22:10:00 0|app | 2021-03-26T22:10:00: info: Connected to MongoDB
22:10:00 0|app | 2021-03-26T22:10:00: info: Listening to port 3000
22:10:00 0|app | 2021-03-26T22:10:00: warn: Unable to connect to email server. Make sure y
ou have configured the SMTP options in .env
```