

# Implementasi Otentikasi Website Node JS Express Menggunakan Passport

**Diterima:**  
21 September 2023  
**Revisi:**  
21 November 2023  
**Terbit:**  
1 Desember 2023

<sup>1\*</sup>Shandy Arshad Busro Cahyono, <sup>2</sup>Sucipto, <sup>3</sup>Rina Firliana  
*<sup>1-3</sup>Universitas Nusantara PGRI Kediri*

**Abstrak**—Pada pembuatan website menggunakan metode linier sequential model dengan tahapan analisis, desain, code dan test. Website yang dibuat berbasis framework Express js Node js dan database MySQL sebagai sarana penyimpanan data. Website yang dibuat didalamnya mengimplementasikan API berarsitektur REST dapat melakukan komunikasi data dengan menggunakan protocol HTTP yang akan mempermudah dalam pengolahan data serta mempermudah dalam memberikan informasi yang dapat diakses oleh banyak orang dimanapun berada dengan menggunakan koneksi internet, dalam melakukan tambah, edit dan hapus data pada website diperlukan autentikasi pengguna untuk membuktikan bahwa pengguna yang hendak login dan melakukan tambah, edit dan hapus data benar-benar merupakan pemilik akun yang sah. Dalam implementasi otentikasi pada website, peneliti menggunakan Node js Passport sebagai middleware yang mudah digunakan untuk aplikasi website berbasis framework Express JS.

**Kata Kunci**— Otentikasi; Node Js; Express Js; Passport

**Abstract**— When creating a website using the linear sequential model method with stages of analysis, design, code and testing. The website created is based on the Express js Node js framework and MySQL database as a means of storing data. The website created therein implements an API with a REST architecture that can communicate data using the HTTP protocol which will make it easier to process data and make it easier to provide information that can be accessed by many people wherever they are using an internet connection, in adding, editing and deleting data on the website. User authentication is required to prove that the user who wants to log in and add, edit and delete data is truly the legitimate account owner. In implementing authentication on websites, researchers use Node js Passport as middleware that is easy to use for website applications based on the Express JS framework.

**Keywords**— Otentikasi; Node Js; Express Js; Passport

This is an open access article under the CC BY-SA License.



---

## Penulis Korespondensi:

Shandy Ashad Busro Chayono  
Sistem Informasi  
Universitas Nusantara PGRI Kediri,  
Email: [shandyabc313@gmail.com](mailto:shandyabc313@gmail.com)

---

## I. PENDAHULUAN

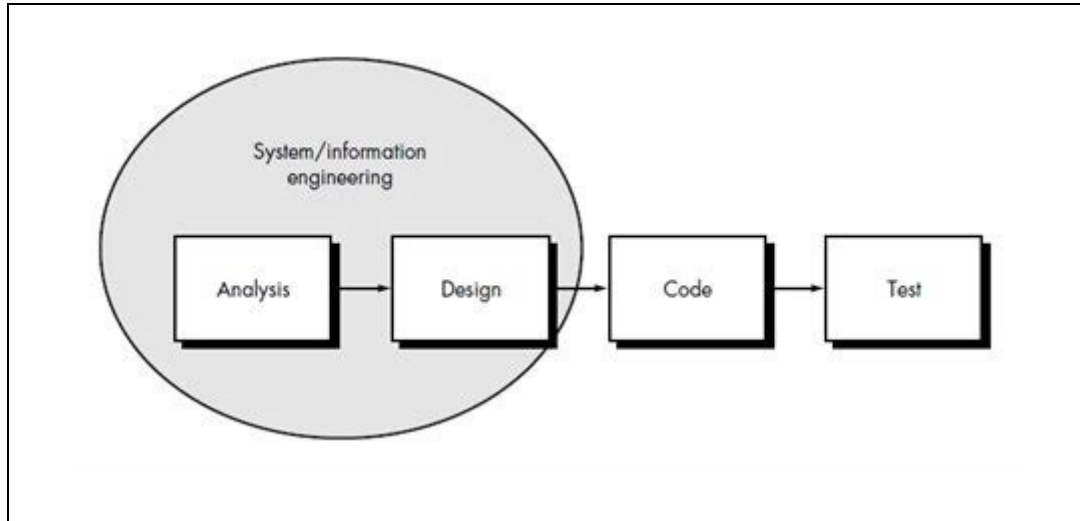
Persebaran tanaman obat dan tanaman langka yang ada di Kabupaten Kediri belum dapat diakses secara umum pada era teknologi saat ini, selain itu pengetahuan tentang tanaman obat masih disampaikan secara lisan dari generasi ke generasi sehingga perlu sebuah media digital untuk membantu mengatasi hal tersebut [1]. Pada era digital hampir seluruh aplikasi misalkan aplikasi berbasis website yang digunakan oleh pengguna saat ini dipastikan memerlukan koneksi ke internet untuk dapat menyediakan layanan yang dibutuhkan pengguna [2]. Website banyak disediakan oleh perorangan, kelompok maupun organisasi yang mana website merupakan sekumpulan halaman yang terdiri dari laman yang saling berhubungan [3]. Website dapat berisikan informasi berupa dalam bentuk text, video, audio maupun gambar [4].

Website dapat bersifat dinamis jika isi informasi dari website selalu berubah serta interaktif dan *website* bersifat statis jika isi dari informasi yang ada pada *website* tetap dan jarang berubah [5]. Dengan adanya sebuah website mampu memberikan informasi menjadi lebih efektif dan mudah diakses oleh pengguna yang membutuhkan suatu informasi dengan cepat [6]. Perubahan informasi yang ada pada website dapat dilakukan oleh pengguna yang benar-benar memiliki hak akses untuk melakukan tambah, edit dan hapus data maka pada website diperlukan otentikasi pengguna [7]. Umumnya otentikasi pengguna menggunakan *username* dan *password* [8]. *Username* dan *password* nantinya akan digunakan untuk proses login ke sistem sebagai tanda pengenalan pengguna [9].

Berdasarkan dari latar belakang tersebut peneliti mencoba mengimplementasikan otentikasi pengguna pada website data tanaman obat menggunakan passport yang berbasis node js, express js dan menggunakan view engine Ejs untuk tampilan pengguna.

## II. METODE

Metode Linear Sequential Model atau sering disebut dengan metode waterfall merupakan salah satu model dari SDLC (Software Development Life Cycle) seri klasik yang digunakan dalam pengembangan sistem informasi [10] [11].



Gambar 1. Metode Linear Sequential Model

Metode sekuensial untuk pengembangan perangkat lunak yang dilakukan secara berurutan seperti pada Gambar 1 [12]. Tahapan Metode Linear Sequential Model sebagai berikut.

1. Analisis Kebutuhan

Pada tahap ini menentukan kebutuhan perangkat lunak yang dibutuhkan oleh user [13]. Tahap ini penulis menganalisa data yang didapatkan oleh anggota yang melakukan pendataan tanaman obat dan langka dilapangan.

2. Desain

Setelah dilakukan analisis langkah selanjutnya membuat desain atau rancangan sistem yang fokus pada perancangan antar muka, struktur data dan sebagainya [14].

3. Code

Tahap ini dilakukan implementasi sesuai desain yang sudah dirancang dengan dilakukannya pengkodean atau pembuatan program aplikasi [15]. Pengkodean menggunakan bahasa JavaScript, Node js ,Framework Express js.

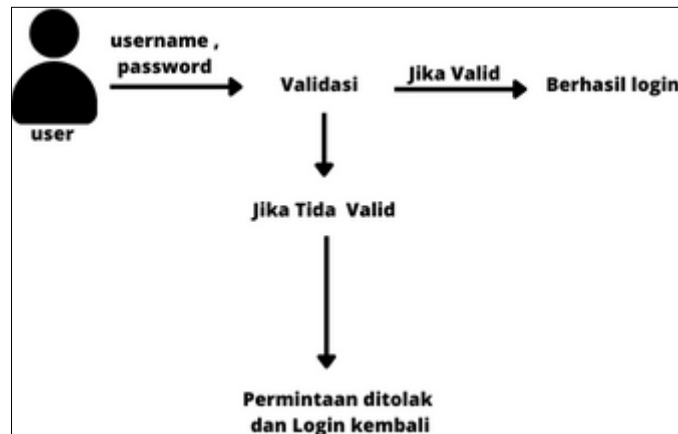
4. Pengujian

Pada tahap ini dilakukan pengujian blackbox untuk memastikan semua bagian sistem dapat bekerja sesuai dengan kebutuhan dan persyaratan yang telah ditetapkan.

### III. HASIL DAN PEMBAHASAN

1. Cara Kerja Autentikasi

Autentikasi merupakan pengaman untuk mencegah pihak-pihak yang tidak memiliki otoritas dalam mengakses sistem bertujuan untuk membuktikan identitas pengguna sebenarnya yang hendak memasuki sistem, salah satu metode untuk otentikasi dapat menggunakan dengan username dan password pengguna seperti pada Gambar 2 berikut



Gambar 2. Cara Kerja Autentikasi[16]–[18]

Pada Gambar 2 user melakukan login ke sistem menggunakan password dan username jika password benar maka user berhasil melakukan login serta akan menampilkan ke halaman utama dan jika username dan password tidak valid maka login gagal dan dikembalikan ke halaman login untuk melakukan login kembali

## 2. Implementasi

### 2.1 Struktur Kerangka Kerja

```
> config 12
> controller 13
> middleware 14
  JS Authjs 15
  JS hadleUpload.js 16
> migrations 17
> models 18
> node_modules 20
> public 21
> routes 22
  JS indexjs 23
  seeders 24
  views 25
  package-lock.json 26
  package.json 27
  JS passport-config.js 28
  JS server.js 29
  dependencies: {
    "bcryptjs": "^2.4.3",
    "body-parser": "^1.20.2",
    "connect-flash": "^0.1.1",
    "ejs": "^3.1.9",
    "express": "^4.18.2",
    "express-flash": "0.0.2",
    "express-session": "^1.17.3",
    "http-errors": "^2.0.0",
    "jsonwebtoken": "^9.0.1",
    "method-override": "^3.0.0",
    "morgan": "^1.10.0",
    "multer": "^1.4.5-lts.1",
    "mysql": "^2.18.1",
    "mysql2": "^2.3.3",
    "nodemon": "^2.0.22",
    "passport": "^0.6.0",
    "passport-local": "^1.0.0",
    "sequelize": "^6.32.1",
    "sequelize-cli": "^6.6.1"
```

Gambar 3. Struktur Kerangka Kerja

Pada Gambar 3 merupakan kerangka kerja setelah instalasi kebutuhan yang diperlukan termasuk instalasi passport, EJS, Express Js. Informasi hasil dari instalasi modul tambahan akan tersimpan pada file package.json.

## 2.2 Mendefinisikan Passport Strategi

```
6 passport.use(  
7   new LocalStrategy(async function (username, password, done) {  
8     try {  
9       const user = await Petugas.findOne({ where: { username: username } });  
10      if (!user) {  
11        return done(null, false, { message: "Username Salah!" });  
12      }  
13      const passVal = await bcryptjs.compare(password, user.password);  
14      if (!passVal) {  
15        return done(null, false, { message: "Password Salah!" });  
16      }  
17      return done(null, user);  
18    } catch (err) {  
19      return done(err);  
20    }  
21  })  
22 );  
23  
24 passport.serializeUser(function (user, done) {  
25   done(null, user.id);  
26 });  
27  
28 passport.deserializeUser(function (id, done) {  
29   Petugas.findById(id).then(function (user) {  
30     done(null, user);  
31   });  
32 });
```

Gambar 4. Passport Strategi Login

Pada Gambar 4 mendefinisikan parameter passport strategi untuk login menggunakan username dan password atau LocalStrategy. secara default LocalStrategy mengharapkan untuk menemukan kredensial pengguna dalam parameter username & password.

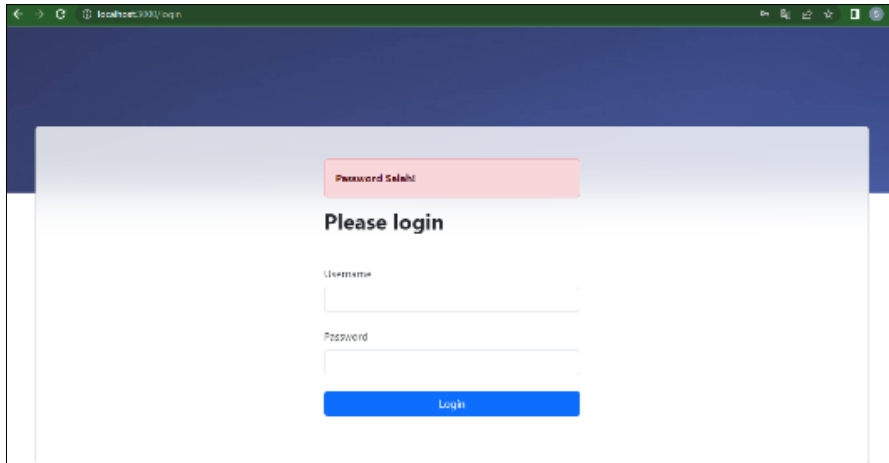
## 2.3 Membuat Route Login

```
app.post(  
  "/login",  
  checkAuthMiddleware.checkNotAuthenticated,  
  passport.authenticate("local", {  
    successRedirect: "/homeAdmin",  
    failureRedirect: "/login",  
    failureFlash: true,  
  })  
);
```

Gambar 5. Route Login

Pada Gambar 5 merupakan route untuk login yang mana user melakukan login menggunakan username dan password jika username dan password user benar maka akan di redirect ke halaman admin dan jika username dan password salah maka akan di redirect ke halaman login untuk melakukan login kembali.

## 2.4 Pengujian



Gambar 6. Login

Pada Gambar 7 tampilan login menggunakan EJS kemudian dilakukan pengujian menggunakan metode black box testing, pada pengujian melakukan login menggunakan username dan password sesuai dengan pendefinisian strategi passport login dan route jika melakukan login dan salah memasukkan username dan password maka akan di redirect ke halaman login dan menampilkan pesan kesalahan.[19]

#### IV. KESIMPULAN

Setelah dilakukan pengujian dapat disimpulkan dapat mengimplementasikan otentikasi sederhana simple dan fleksibel menggunakan passport sebagai middleware yang mudah digunakan untuk aplikasi website berbasis framework Express JS. untuk membuktikan bahwa pengguna yang hendak login dan melakukan tambah, edit dan hapus data benar-benar merupakan pemilik akun yang sah.

#### DAFTAR PUSTAKA

- [1] M. Huda, T. I. Sulistiyowati, P. R. Primandiri, and A. M. Santoso, "Kajian Etnobotani Tanaman Obat di Desa Jugo Kecamatan Mojo Kabupaten Kediri," *Prosiding Seminar Nasional Kesehatan, Sains dan Pembelajaran*, vol. 2, no. 1, pp. 493–502, 2022.
- [2] I. G. N. Ady Kusuma, "Perancangan Simple Stateless Autentikasi Dan Otorisasi Layanan Rest-Api Berbasis Protokol Http," *Jurnal Manajemen Informatika dan Sistem Informasi*, vol. 4, no. 1, p. 78, 2021, doi: 10.36595/misi.v4i1.325.
- [3] Rina Firliana, Fatrkur Rhoman, and Rolistian Wikky Purwinanto, "Perancangan Sistem Informasi Absensi Dosen Dengan Validasi Mahasiswa Berbasis Web," *Jurnal Sains dan Informatika*, vol. 4, 2018.

- [4] M. Huda, "Website sebagai Media Informasi dan Bisnis," *Journal of Community Service and Empowerment*, vol. 1, no. 1, pp. 56–68, 2020.
- [5] R. B. B. Sumantri, W. Setiawan, and D. N. Triwibowo, "Rancang Bangun Aplikasi Media Jasa Desain Logo Dengan Metode Waterfall Berbasis Website," *METHOMIKA Jurnal Manajemen Informatika dan Komputerisasi Akuntansi*, vol. 6, no. 6, pp. 157–163, 2022, doi: 10.46880/jmika.vol6no2.pp157-163.
- [6] W. Andriyan, S. S. Septiawan, and A. Aulya, "Perancangan Website sebagai Media Informasi dan Peningkatan Citra Pada SMK Dewi Sartika Tangerang," *Jurnal Teknologi Terpadu*, vol. 6, no. 2, pp. 79–88, 2020, doi: 10.54914/jtt.v6i2.289.
- [7] N. P. Lestari, Y. Durachman, S. Watini, and S. Millah, "Manajemen Kontrol Akses Berbasis Blockchain untuk Pendidikan Online Terdesentralisasi," *Technomedia Journal*, vol. 6, no. 1, pp. 111–123, 2021, doi: 10.33050/tmj.v6i1.1682.
- [8] M. D. Cookson and P. M. R. Stirk, "Aplikasi Pencatatan Inventori Berbasis Website Dengan Skema Autentikasi Dan Otorisasi Stateless Sederhana," *Journal of Innovation Research and Knowledge*, vol. 1, no. 9, pp. 1111–1120, 2019.
- [9] F. Wibowo, M. Hasbi, and Y. Arafat, "Integrasi Otentikasi Aplikasi Berbasis Web Menggunakan RADIUS (Studi Kasus: Website E-Learning Jurusan Teknik Elektro POLNEP)," *Jurnal ELIT*, vol. 1, no. 2, pp. 1–12, 2020, doi: 10.31573/elit.v1i2.51.
- [10] S. F. Ula, "Sistem Informasi Kepegawaian Dinas Pendidikan menggunakan metode WATERFALL," *Jurnal Teknik Informatika UNIKA Santo Thomas*, vol. 05, no. 21, pp. 403–411, 2021, doi: 10.54367/jtiust.v6i2.1569.
- [11] M. A. Dharmawan, R. Indriati, and S. Sucipto, "Implementasi Sistem Informasi Tugas Akhir Menggunakan Metode Classic Life Cycle," *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, vol. 3, no. 1, pp. 151–154, 2019.
- [12] S. Sucipto, E. K. Dewi, N. C. Resti, and I. H. Santi, "Improving The Performance of Alumni Achievement Assessment by Integrating Website-Based Tracer Study Information Systems and Telegram API," *Teknik*, vol. 41, no. 1, pp. 72–77, 2020, doi: 10.14710/teknik.v41i1.25307.
- [13] D. Puspitasari, "Sistem Informasi Perpustakaan Sekolah Berbasis Web," *Jurnal Pilar Nusa Mandiri Vol. XII*, vol. 12, no. 2, pp. 227–240, 2016.
- [14] F. Ardiansyah, A. S. Wardani, and S. Sucipto, "Rancang Bangun Company Profile Pusat Pelayanan Terpadu Perlindungan Perempuan dan Anak Berbasis Website," *JSITIK: Jurnal Sistem Informasi dan Teknologi Informasi Komputer*, vol. 1, no. 2, pp. 124–136, 2023, doi: 10.53624/jsitik.v1i2.176.
- [15] B. Fachri and R. W. Surbakti, "Perancangan Sistem Dan Desain Undangan Digital Menggunakan Metode Waterfall Berbasis Website (Studi Kasus: Asco Jaya)," *Journal of Science and Social Research*, vol. 4, no. 3, p. 263, 2021, doi: 10.54314/jssr.v4i3.692.
- [16] P. Peddada, "API authentication," US Patent 10,412,068, 2019, [Online]. Available: <https://patents.google.com/patent/US10412068B2/en>

- [17] S. Preibisch and S. Preibisch, "API Authentication and Authorization," *API Development: A Practical Guide for Business ...*, 2018, doi: 10.1007/978-1-4842-4140-0\_5.
- [18] S. Sucipto and J. Karaman, "Integration of Legalization Information System Web-Based using Shipping API and Telegram API," *JUITA: Jurnal Informatika*, vol. 8, no. 2, pp. 131–139, 2020.
- [19] Sucipto et al., "Hidden Treasures of Kediri's Medicinal Plants: A Collaborative Effort to Map and Validate Authentic Information Using Innovative QR Code Security and Cryptography," *IOP Conf Ser Earth Environ Sci*, vol. 1242, no. 1, p. 012036, Sep. 2023, doi: 10.1088/1755-1315/1242/1/012036.