

README EDUNOOK ITB

- Deskripsi aplikasi: Edunook ITB adalah sebuah aplikasi web yang menyediakan rekomendasi tempat belajar di sekitar tiga kampus utama Institut Teknologi Bandung (ITB): Ganesha, Jatinangor, dan Cirebon. Aplikasi ini dirancang untuk membantu mahasiswa dan masyarakat sekitar dalam menemukan tempat yang nyaman dan mendukung untuk belajar, baik di dalam kampus maupun di sekitarnya. Fitur-fitur :
 - **Filter Pencarian:** Memungkinkan pengguna untuk menyaring tempat belajar berdasarkan kriteria seperti suasana (tenang, kreatif, sosial), fasilitas (Wi-Fi, colokan listrik, ruang diskusi), dan jarak dari kampus.
 - **Ulasan dan Penilaian:** Menyediakan fitur bagi pengguna untuk memberikan ulasan dan penilaian tempat belajar, sehingga membantu pengguna lain dalam memilih lokasi yang tepat.
 - **Rekomendasi Tempat:** Menyediakan rekomendasi tempat belajar berdasarkan tren, popularitas, dan kebutuhan pengguna.
 - **Peta Interaktif:** Fitur peta yang dapat menunjukkan jarak serta arah pengguna menuju ke tempat belajar tersebut
- Teknologi, bahasa, dan *framework* yang digunakan.
 - **Frontend:** HTML, CSS, JavaScript, dan React.js
 - **Backend:** Node.js dengan Express.js
 - **Database:** MongoDB
 - **Peta:** Google Maps API
- Struktur program.

```
const mongoose = require("mongoose");
```

```
const placeSchema = new mongoose.Schema({  
  nama: {type: String,required:true},  
  lokasi: {type: String,required:true},  
  buka: {type: String,required:true},
```

```
tutup: {type: String,required:true},
hariTutup: {type: [String],required:true},
kursi: {type: String,required:true},
meja: {type: String,required:true},
listrik: {type: Boolean,required:true},
ac: {type: Boolean,required:true},
bebasRokok: {type: Boolean,required:true},
konduSif: {type: Boolean,required:true},
lepasAlasKaki: {type: Boolean,required:true},
makan: {type: Boolean,required:true},
minum: {type: Boolean,required:true},
wifi: {type: Boolean,required:true},
fasilitas: {type: [String],required:true},
img: {type: String,required:true}
});
```

```
module.exports = mongoose.model("place", placeSchema);
```

```
const router = require("express").Router();
```

```
const Place = require("../models/Place");
```

```
router.get("/places", async (req,res) => {
  try {
    const page = parseInt(req.query.page) - 1 || 0;
    const limit = parseInt(req.query.limit) || 5;
    const search = req.query.search || "";
    let sort = req.query.sort || "rating";
    let fasilitas = req.query.fasilitas || "All";

    const facilities =[
      "Indoor",
```

```
"Semi Outdoor",  
"Outdoor",  
"Luas",  
"Sumber Listrik",  
"Wifi",  
"Bebas Asap Rokok",  
"Kondusif",  
"AC",  
"Boleh membawa makanan",  
"Boleh membawa minuman",  
"Bisa untuk diskusi",  
];
```

```
fasilitas === "All"  
? (fasilitas = [...facilities])  
: (fasilitas= req.query.fasilitas.split(","));  
req.query.sort ? (sort=req.query.sort.split(",")) : (sort=[sort]);
```

```
let sortBy={};  
if (sort[1]) {  
  sortBy[sort[0]] = sort[1];  
} else {  
  sortBy[sort[0]] = "asc";  
}
```

```
const places = await Place.find({nama: {$regex: search, $options: 'i'}})  
  .where("fasilitas").in([...fasilitas])  
  .sort(sortBy)  
  .skip(page * limit)
```

```

        .limit(limit);
const total = await Place.countDocuments({
  nama: {$regex:search,$options:'i'},
});
const response = {
  error:false,
  total,
  page:page+1,
  limit,
  places,
};
res.status(200).json(response);

} catch (error) {
  console.log(error);
  res.status(500).json({error:true, message:"Internal Server Error"});
}
});

```

```

module.exports = router;

```

- Cara menjalankan program: mulai dari instalasi requirement hingga langkah-langkah menjalankan aplikasi.
 1. Instalasi Requirement clone repository
 2. Setup Environment
 3. Install Dependencies
 4. Menjalankan aplikasi
 5. Deployment

- Anggota kelompok beserta *role*-nya :

- | Kevin | Product Manager |
- | Sabil | Data Science |
- | Inggried | Data Science |
- | Leon | Data Science
- | Zaka | Software Developer |
- | Orvin | Software Developer |
- | Edo | Software Developer |
- | Julius | Software Developer |
- | Mahesa | Software Developer |
- | Luckman | Software Developer |
- | Gabriela | UI/UX |
- Referensi belajar bila ada.
 - Channel youtube procademy