





# Implementing Social News Web App

In this coding project, you will extend the previous project and implement a Social News Web App out of it. Your project must follow all the rules stated in the previous lesson. Starter HTML and CSS, and JavaScript code is provided in the coding playground below.

Since this project requires a **Node.js** server, so you can implement this web app on your local host. A solution is provided below for your reference, but try to solve it on your first. Good luck!

 index.js	 publiclink.js	 link.js	 publiclink.css
--	---	---	--

```
const express = require("express");
const multer = require("multer");
const Link = require("../modules/link.js");

const app = express();
const upload = multer();

// Enable CORS (see https://enable-cors.org/server_expressjs.html)
app.use((req, res, next) => {
  res.header("Access-Control-Allow-Origin", "*");
  res.header(
    "Access-Control-Allow-Headers",
    "Origin, X-Requested-With, Content-Type, Accept"
  );
  next();
});

// Serve content of the "public" subfolder directly
app.use(express.static("public"));

// Initial links array
const links = [];
links.push(new Link("Hacker News", "https://news.ycombinator.com", "Baptiste"));
links.push(new Link("Reddit", "https://reddit.com", "Thomas"));
links.push(new Link("Boing Boing", "boingboing.net", "Daniel"));

// Routes

app.get("/", (request, response) => {
  response.sendFile(`${__dirname}/views/index.html`);
});

app.get("/api/links", (request, response) => {
  response.json(links);
});
```

```
app.post("/links", upload.array(), (request, response) => {  
  // Create new link object  
  const title = request.body.title;  
  
  const url = request.body.url;  
  const author = request.body.author;  
  const link = new Link(title, url, author);  
  // Add new link to the beginning of the list  
  links.unshift(link);  
  // Send back newly created link as JSON  
  response.json(link);  
});  
  
// Start listening to incoming requests  
// If process.env.PORT is not defined, port number 3000 is used  
const listener = app.listen(process.env.PORT || 3000, () => {  
  console.log(`Your app is listening on port ${listener.address().port}`);  
});
```