

1.server.js

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

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

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

```
// Port to listen on
```

```
const PORT = 3000;
```

```
// Helper function to serve HTML files asynchronously
```

```
const serveFile = (filePath, contentType, response, statusCode = 200) => {
```

```
  fs.readFile(filePath, (err, content) => {
```

```
    if (err) {
```

```
      response.writeHead(500, { 'Content-Type': 'text/plain' });
```

```
      response.end('500 - Internal Server Error');
```

```
    } else {
```

```
      response.writeHead(statusCode, { 'Content-Type': contentType });
```

```
      response.end(content);
```

```
    }
```

```
  });
```

```
};
```

```
// Create server
```

```
const server = http.createServer((req, res) => {
```

```
  console.log(`Requested URL: ${req.url}`);
```

```
  if (req.url === '/home' || req.url === '/') {
```

```
    serveFile(path.join(__dirname, 'pages', 'home.html'), 'text/html', res);
```

```
  } else if (req.url === '/about') {
```

```
    serveFile(path.join(__dirname, 'pages', 'about.html'), 'text/html', res);
  } else if (req.url === '/contact') {
    serveFile(path.join(__dirname, 'pages', 'contact.html'), 'text/html', res);
  }
  else {
    serveFile(path.join(__dirname, 'pages', '404.html'), 'text/html', res, 404);
  }
});

// Start server
server.listen(PORT, () => {
  console.log(`Server running at http://localhost:${PORT}/`);
});
```

2.css/style.css

```
body {
  font-family: Arial, sans-serif;
  background-color: #f0f0f0;
  text-align: center;
  padding: 50px;
}
h1 {
  color: #333;
}
p {
  color: #555;
}
```

3.pages/home.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Home</title>
  <link rel="stylesheet" href="/css/style.css">
</head>
<body>
  <h1>Welcome to the Home Page</h1>
  <p>This is the home route of our Node.js server.</p>
</body>
</html>
```

4.pages/about.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>About</title>
  <link rel="stylesheet" href="/css/style.css">
</head>
<body>
  <h1>About Us</h1>
  <p>This is the about page of our Node.js server.</p>
</body>
</html>
```

5. pages/contact.html

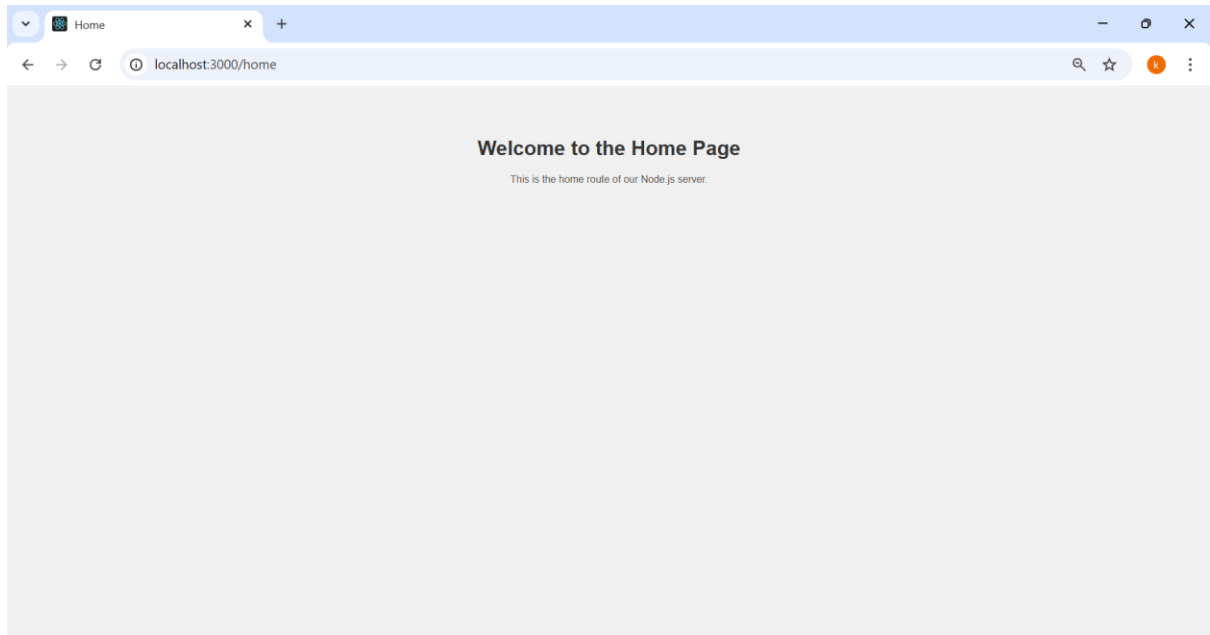
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Contact</title>
  <link rel="stylesheet" href="/css/style.css">
</head>
<body>
  <h1>Contact Us</h1>
  <p>This is the contact page of our Node.js server.</p>
</body>
</html>
```

6. pages/404.html

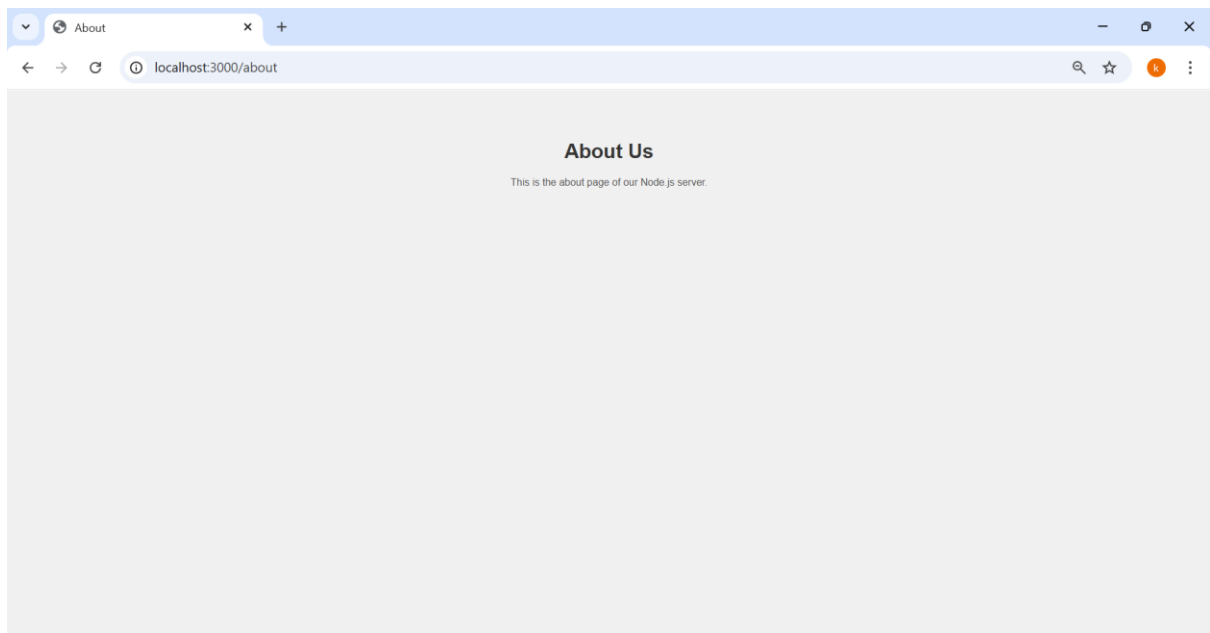
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>404 Not Found</title>
  <link rel="stylesheet" href="/css/style.css">
</head>
<body>
  <h1>404 - Page Not Found</h1>
  <p>Oops! The page you are looking for does not exist.</p>
</body>
</html>
```

7.Images

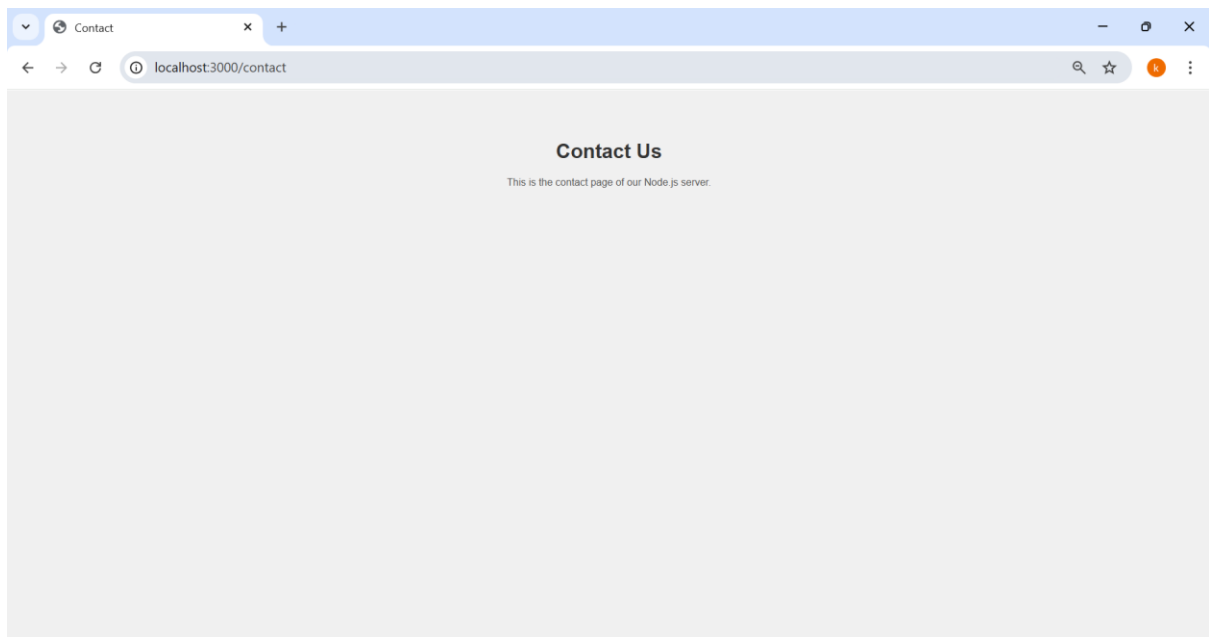
1.Home Page



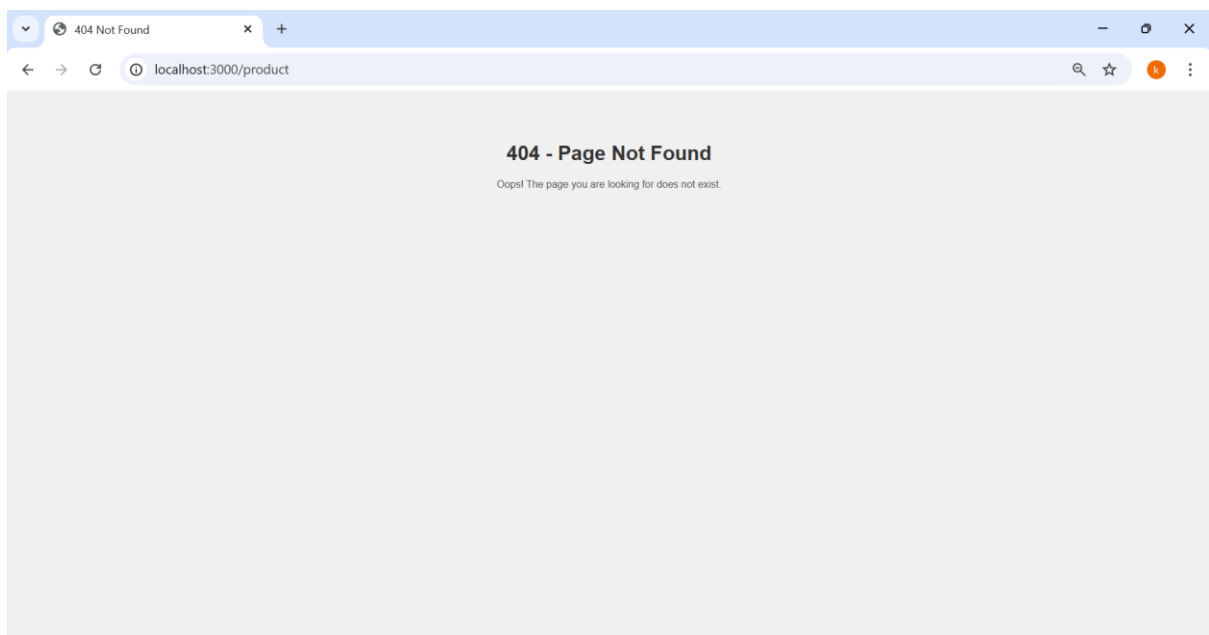
2.About Page



3.Contact Page



4. 404 Page (If url is mismatched)



Brief Explanation of How the Server Works

This Node.js server uses the built-in `http` module to handle incoming HTTP requests. It serves different HTML pages based on the requested URL path, implementing simple routing:

1. Server Creation:

- The server is created using `http.createServer()` and listens on port 3000.

2. Routing:

- The server checks the requested URL (`req.url`) and serves the corresponding HTML page:
 - `/home` → `home.html`
 - `/about` → `about.html`
 - `/contact` → `contact.html`
- Any unknown route returns a custom 404 page with a 404 HTTP status code.

3. File Handling:

- HTML and CSS files are read asynchronously using `fs.readFile()` for better performance.
- CSS is served for styling, and all pages link to the same stylesheet.

4. HTTP Status Codes:

- Valid pages return **200 OK**.
- Invalid routes return **404 Not Found**.
- Server errors return **500 Internal Server Error**.

5. Testing:

- When the server is running, users can navigate to `http://localhost:3000/home`, `/about`, or `/contact` to see the corresponding pages.
- Any invalid URL automatically shows the 404 page.