



Start.ejs

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" />
  <link rel="stylesheet" href="./css/index.css">
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>

<body>
```

End.ejs

```
</body>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha1/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-
w76AqPfDkMBDXo30jS1Sgez6pr3x5MlQ1ZAGC+nuZB+EYdgRZgiwxhTBTkF7CXvN"
  crossorigin="anonymous"></script>

</html>
```

Engineers.ejs

```
<%- include('start') -%>
<%- include('header') -%>
  <main class="container">
    <div class="mt-5 mb-4">
      <div class="mb-4 gap-3">
        <h1>Star Engineer</h1>
        <div id="cardCarousel" class="carousel slide" data-bs-ride="carousel">
          <div class="carousel-inner">
            <% stars.forEach(e=>{%>

              <div class="carousel-item active">
                <div class="row justify-content-center">
                  <div class="col-md-4">
                    <div class="card">
                      <img src= '<%= e.imgurl%>' class="card-img-top" alt="Card 1">
                      <div class="card-body">
                        <h5 class="card-title">Sarah Chen</h5>
                        <p class="card-text"><%= e.description%></p>
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            <%}) %>
          </div>
          <button class="carousel-control-prev" type="button" data-bs-
target="#cardCarousel"
            data-bs-slide="prev">
            <span class="carousel-control-prev-icon" aria-hidden="true"></span>
            <span class="visually-hidden">Previous</span>
          </button>
          <button class="carousel-control-next" type="button" data-bs-
target="#cardCarousel"
            data-bs-slide="next">
            <span class="carousel-control-next-icon" aria-hidden="true"></span>
            <span class="visually-hidden">Next</span>
          </button>
        </div>
      </div>
    </div>
```

```

<div class="row mb-4">
  <div class="mb-4 row">
    <h1 class="col-9">Engineer List</h1>
    <input type="text" class="col-3 rounded" placeholder="input the description
of the engineer">
  </div>
  <div class="mb-4 btn-group" role="group" aria-label="Basic example">
    <div class="btn-group col-3" role="group">
      <button type="button" class="btn text-info-emphasis text-bg-light border
dropdown-toggle"
        data-bs-toggle="dropdown" aria-expanded="false">Type</button>
      <ul class="dropdown-menu w-100 text-center">
        <li><a class="dropdown-item"
href="?type=electrician">Electrician</a></li>
        <li><a class="dropdown-item"
href="?type=carpenter">Carpenter</a></li>
      </ul>
    </div>
    <a type="button" class="btn text-info-emphasis text-bg-light border col-3"
href="?field=area">Area</a>
    <a type="button"
      class="btn text-info-emphasis text-bg-light border col-3"
href="?field=price">Saticification</a>
    <a type="button" class="btn text-info-emphasis text-bg-light border col-3"
href="?field=price">Price</a>
  </div>

  <% engineers.forEach(e=>{ %>

    <div class="mb-4 col-lg-4 col-md-6 col-sm-10">
      <div class="card col-lg-6" style="width: 18rem;">
        <img src='<%= e.imgurl%>' class="card-img-top" alt="...">
        <div class="card-body">
          <p class="card-text"><%= e.description%></p>
        </div>
      </div>
    </div>
    <%}} %>
  </div>
</div class="row">
  <div class="offset-lg-5 offset-md-4 offset-sm-4 col-lg-4 col-md-4 col-sm-4">
    <nav<ul class="pagination">
      <li class="page-item <%= parseInt(currentPage) === 1 ? 'disabled' : " %>">
        <a class="page-link" href="?page=<%= parseInt(currentPage) -
1 %>">Previous</a>
      </li>

      <% for (let i = parseInt(currentPage); i <= parseInt(totalPages) && i <=
(parseInt(currentPage) + 5); i++) { %>
        <li class="page-item <%= i === parseInt(currentPage) ? 'active' : " %>">
          <a class="page-link" href="?page=<%= i %>">

```

```

        <%= i %>
      </a>
    </li>
  <% } %>

  <li
    class="page-item <%= parseInt(currentPage) ===
parseInt(totalPages) ? 'disabled' : " %>">
    <a class="page-link"
      href="?page=<%= parseInt(currentPage) + 1 %>">Next</a>
    </li>
  </ul>
</nav>
</div>
</div>
</div>
</main>
<%- include('footer') -%>
<%- include('end') -%>

```

Footer.ejs

```

<footer class="container">
  <div class="text-center">&copy; Home Repaire <p><em>Help every homeover solve
home maintenance problems</em></p>
  </div>
</footer>

```

Header.ejs

```

<header class="container">
  <div class="row justify-content-between align-items-center mt-5 mb-5 flex-nowrap">
    <div class="col-auto">
      <h1 class="display-5 fw-bold text-body-emphasis" style="white-space:
nowrap;">Home Repair</h1>
    </div>
    <% if (!login) { %>
      <div class="col-auto">
        <div class="d-flex gap-3">
          <a class="btn btn-primary" href="/login">Login</a>
          <a class="btn btn-primary" href="/register">Register</a>
        </div>
      </div>
    <% }else{ %>

    <% } %>
  </div>

  <nav class="navbar sticky-top navbar-expand-lg rounded" aria-label="Thirteenth navbar
example">
    <div class="container-fluid">

```

```

        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarsExample11"
        aria-controls="navbarsExample11" aria-expanded="false" aria-label="Toggle
navigation">
        <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse d-lg-flex justify-content-center"
id="navbarsExample11">
        <ul class="navbar-nav col-lg-6 justify-content-lg-center">
            <li class="nav-item col-lg-3">
                <a class="nav-link active" aria-current="page"
href="/introduction">Introduction</a>
            </li>
            <li class="nav-item col-lg-3">
                <a class="nav-link" href="/engineers">Engineers</a>
            </li>
            <li class="nav-item col-lg-3">
                <a class="nav-link" href="/homeIssues">Home Issues</a>
            </li>
            <li class="nav-item col-lg-3">
                <a class="nav-link" href="/news">News</a>
            </li>
        </ul>
    </div>
</div>
</nav>
</header>

```

homeIssues.ejs

```

<%- include('start') -%>
<%- include('header') -%>
<link rel="stylesheet" href="../css/home_issues.css">
<main>
    <div class="container">
        <h1 class="mb-4">Emergency</h1>
        <% emergencyIssues.forEach(e=>{ %>
            <div class="row mb-4 text-center">
                <div class="col-lg-6">
                    <img class="rounded mx-auto d-block" style="max-width: 100%;" src='<%=
e.imgurl %>'
                    alt="house info">
                </div>
                <div class="col-lg-6">
                    <p class="text-center">
                        <%= e.description %>
                    </p>
                </div>
            </div>
        </div>
    </div>

```

```

<% }} %>

<h1>Urgent List</h1>
<div class="row">
  <% urgencyIssues.forEach(e=>{ %>
    <div class="my-4 col-lg-6 col-md-6 col-sm-10">
      <div class="card col-lg-6" style="width: 18rem;">
        <img src='<%= e.imgurl %>' class="card-img-top" alt="...">
        <div class="card-body">
          <p class="card-text">
            <%= e.description %>
          </p>
        </div>
      </div>
    </div>
  <% }} %>
</div>
<div class="row">
  <div class="offset-lg-5 offset-md-4 offset-sm-4 col-lg-4 col-md-4 col-sm-4">
    <nav aria-label="Page navigation example">
      <ul class="pagination">
        <li class="page-item <%= parseInt(currentPage) === 1 ? 'disabled' :
" %>">
          <a class="page-link"
            href="?page=<%= parseInt(currentPage) - 1 %>">Previous</a>
        </li>
        <% for (let i=1 ; i <=totalPages; i++) { %>
          <li class="page-item <%= i === parseInt(currentPage) ? 'active' :
" %>">
            <a class="page-link" href="?page=<%= i %>">
              <%= i %>
            </a>
          </li>
          <% } %>
          <li
            class="page-item <%= parseInt(currentPage) ===
parseInt(totalPages) ? 'disabled' : " %>">
            <a class="page-link"
              href="?page=<%= parseInt(currentPage)+1 %>">Next</a>
          </li>
        </ul>
      </nav>
    </div>
  </div>
</div>
</main>
<%- include('footer') -%>
<%- include('end') -%>

```

Introduction.ejs

```
<%- include('start') -%>
```

```

<%- include('header') -%>
<main class="container">
  <section class="py-5 text-center container">
    <div class="row py-lg-5">
      <h3 class="text-center fw-normal text-muted mb-3">Every year, thousands of
homes are damaged and
      need
      repairs. Our vision is to
      help homeowners get their
      houses repaired as quickly as possible, with the highest quality.</h3>
    </div>
  </section>
  <div class="row mb-4 mx-5">
    
  </div>
  <div class="row mb-4 mx-5">
    
  </div>
</main>
<%- include('footer') -%>
<%- include('end') -%>

```

Login.ejs

```

<%- include('start') -%>
<header class="container">
  <div class="row text-center mt-5 mb-5">
    <div class="offset-sm-2 col-sm-8 col-md-8 col-lg-8">
      <h1 class="display-5 fw-bold text-body-emphasis">Home Repair</h1>
    </div>
  </div>
</header>
<main>
  <div class="offset-2 col-8">
    <div class="col-lg-6 col-xxl-4 my-5 mx-auto">
      <div class="row gap-2">
        <a class="col btn btn-primary" href="/introduction">Home</a>
        <a class="col btn btn-primary" href="/register">register</a>
      </div>
    </div>
    <form id="loginForm" class="needs-validation" novalidate>
      <div class="row my-5">
        <label for="emailOrPhone" class="form-label">Email address/Phone</label>
        <input type="text" class="form-control" id="emailOrPhone" aria-
describedby="userInputHelp"
        placeholder="We'll never share your email/phone with anyone else."
onfocus="clearTips('login')"
        required>

```

```

        <div class="valid-feedback">
            Looks good!
        </div>
        <div class="invalid-feedback">
            Please input an email or a phone
        </div>
    </div>
    <div class="row my-5">
        <label for="password" class="form-label">Password</label>
        <input type="password" class="form-control" onfocus="clearTips('pwd')"
id="password">
        <div class="valid-feedback">
            Looks good!
        </div>
        <div class="invalid-feedback">
            Please input your password!
        </div>
    </div>
    <div class="row gap-3">
        <button type="button" class="col btn btn-primary"
onclick="login()">Submit</button>
        <button class="col btn btn-info" onclick="reset()">Reset</button>
    </div>
</form>
</div>
</main>
</body>

<script>
    function validateField(fieldTag) {
        fieldTag.classList.remove("is-valid", "is-invalid");

        if (!fieldTag.value.trim()) {
            fieldTag.classList.add("is-invalid");
        } else {
            fieldTag.classList.add("is-valid");
        }
    }

    function login() {
        let usernameTag = document.getElementById('emailOrPhone')
        let pwdTag = document.getElementById('password')

        validateField(usernameTag);
        validateField(pwdTag);

        let user = {
            name: usernameTag.value,
            password: pwdTag.value
        }
    }

```



```

    fetch('/login/checkUser', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json'
      },
      body: JSON.stringify(user)
    })
    .then(res => {
      if (!res.ok) {
        throw new Error('Network response was not ok');
      }
      return res.json()
    })
    .then(data => {
      sessionStorage.setItem('userData', JSON.stringify(data))
      window.location.href = '/introduction'
    })
    .catch(err => {
      if (err) {
        console.log(err.msg)
      }
      console.log('err: ', err)
    })
  }

function clearTips(methond) {
  let usernameTag = document.getElementById('emailOrPhone')
  let pwdTag = document.getElementById('password')
  if ('login' == methond) {
    usernameTag.classList.remove("is-valid", "is-invalid");
  } else {
    pwdTag.classList.remove("is-valid", "is-invalid");
  }
}

function reset() {
  let usernameTag = document.getElementById('emailOrPhone')
  let pwdTag = document.getElementById('password')

  usernameTag.classList.remove("is-valid", "is-invalid");
  pwdTag.classList.remove("is-valid", "is-invalid");

  usernameTag.value = ""
  pwdTag.value = ""
}
</script>
<%- include('end') -%>

```

News.ejs

```

<%- include('start') -%>
<%- include('header') -%>

```

```

<main>
  <div class="container mt-5">
    <% if(news != null && news.length > 0) { %>
      <% news.forEach(e=> { %>
        <div class="row mb-4">
          <div class="col-lg-8">
            <img class="rounded mx-auto d-block" style="max-width: 100%;" src=<%=
e.imageUrl %>
              alt="news 1">
            </div>
            <div class="col-lg-4">
              <p class="text-center">
                <%= e.description %>
              </p>
            </div>
          </div>
        <% }); %>

        <div class="row">
          <div class="offset-lg-5 offset-md-4 offset-sm-4 col-lg-4 col-md-4 col-sm-4">
            <nav aria-label="Page navigation example" class="my-4">
              <ul class="pagination">
                <li class="page-item <%= parseInt(currentPage) === 1 ? 'disabled' : '' %>">
                  <a class="page-link" href="?page=<%= parseInt(currentPage) -
1 %>">Previous</a>
                </li>

                <% for (let i = parseInt(currentPage); i <= parseInt(totalPages) && i <=
(parseInt(currentPage) + 5); i++) { %>
                  <li class="page-item <%= i === parseInt(currentPage) ? 'active' : '' %>">
                    <a class="page-link" href="?page=<%= i %>">
                      <%= i %>
                    </a>
                  </li>
                <% } %>

                <li
                  class="page-item <%= parseInt(currentPage) ===
parseInt(totalPages) ? 'disabled' : '' %>">
                  <a class="page-link"
                    href="?page=<%= parseInt(currentPage) + 1 %>">Next</a>
                </li>
              </ul>
            </nav>
          </div>
        </div>
      <% } %>
    </div>
  </main>
  <%- include('footer') -%>
  <%- include('end') -%>

```

Register.ejs

```
<%- include('start') -%>
<header class="container">
  <div class="row text-center mt-5 mb-5">
    <div class="offset-sm-2 col-sm-8 col-md-8 col-lg-8">
      <h1 class="display-5 fw-bold text-body-emphasis">Home Repair</h1>
    </div>
  </div>
</header>
<main>
  <div class="offset-3 col-6">
    <div class="col-lg-6 col-xxl-4 my-5 mx-auto">
      <div class="row gap-2">
        <a class="col btn btn-primary" href="/introduction">Home</a>
        <a class="col btn btn-primary" href="/login">login</a>
      </div>
    </div>
    <form class="needs-validation" novalidate>
      <div class="mb-3">
        <label for="name" class="form-label">Name</label>
        <input type="text" class="form-control" id="name" aria-
describedby="userInputHelp"
placeholder="We'll never share your email/phone with anyone else."
required
onfocus="clearTips(this)" onblur="checkInput(this)">
        <div class="valid-feedback">
          Looks good!
        </div>
        <div class="invalid-feedback">
          username must be inputed
        </div>
      </div>
      <div class="mb-3">
        <label for="password" class="form-label">Password</label>
        <input type="password" class="form-control" id="password"
onfocus="clearTips(this)"
onblur="checkInput(this)">
        <div class="valid-feedback">
          Looks good!
        </div>
        <div class="invalid-feedback">
          password must be inputed
        </div>
      </div>
      <div class="mb-3">
        <label for="phone" class="form-label">Phone</label>
        <input type="phone" class="form-control" id="phone"
onfocus="clearTips(this)"
```

```

        onblur="checkInput(this)">
    <div class="valid-feedback">
        Looks good!
    </div>
    <div class="invalid-feedback">
        Phone must be inputed
    </div>
</div>
<div class="mb-3">
    <label for="email" class="form-label">email</label>
    <input type="email" class="form-control" id="email" onfocus="clearTips(this)"
        onblur="checkInput(this)">
    <div class="valid-feedback">
        Looks good!
    </div>
    <div class="invalid-feedback">
        Email must be inputed
    </div>
</div>
<div class="row">
    <div class="col-lg-6 col-sm-12 mb-3">
        <label for="state" class="form-label">state</label>
        <input type="text" class="form-control" id="state">
    </div>
    <div class="col-lg-6 col-sm-12 mb-3">
        <label for="subcode" class="form-label">Subcode</label>
        <input type="text" class="form-control" id="subcode">
    </div>
</div>
<div class="row gap-5">
    <button type="button" class="col btn btn-primary"
onclick="formCheck()">Register</button>
    <button type="button" class="col btn btn-info"
onclick="toHome()">Cancel</button>
</div>
</form>
</div>
</main>
</body>
<script>
    function toHome() {
        window.location.href = './introduction'
    }

    function checkNullAndEmpty(tag) {
        if (tag.val() == null || tag.val() == "") {
            tag.addClass("is-invalid")
            tag.removeClass("is-valid")
            return true
        } else {
            tag.addClass("is-valid")

```

```

        tag.removeClass("is-invalid")
    }
    return false
}

function formCheck() {
    let name = $('#name')
    let password = $('#password')
    let phone = $('#phone')
    let email = $('#email')
    let state = $('#state')
    let subcode = $('#subcode')

    if (checkNullAndEmpty(name)
        || checkNullAndEmpty(password)
        || checkNullAndEmpty(phone)
        || checkNullAndEmpty(email)
        || checkNullAndEmpty(state)
        || checkNullAndEmpty(subcode)) {
        return
    }

    let user = {
        name: name.val(),
        password: password.val(),
        phone: phone.val(),
        email: email.val(),
        state: state.val(),
        subcode: subcode.val()
    }

    fetch('/register/addUser',{
        method: 'POST',
        headers:{
            'Content-Type': 'application/json'
        },
        body: JSON.stringify(user)
    })
    .then(res=>res.json())
    .then(data=>{
        console.log('success: ',data)
        window.location.href = '/login'
    })
    .catch(error=>{
        if(error){
            alert(error.msg)
        }
        console.log('error: ',error)
    })
}

```

```
function clearTips(tag) {
  tag.classList.remove("is-valid", "is-invalid");
}

function checkInput(tag) {
  if (tag.value == null || tag.value == "") {
    tag.classList.add("is-invalid")
  } else {
    tag.classList.add("is-valid")
  }
}
</script>

<%- include('end') -%>
```

Js

Index.js

```
const express = require('express')
const morgan = require('morgan')
const path = require('path')
const session = require('express-session')

const port = 3000
const app = express()

app.use(express.static('public'))
app.use(express.json())
app.use(express.urlencoded(false))
app.use(morgan('common'))

app.use(session({
  secret: 'Chris 224212855',
  resave: false,
  saveUninitialized: true,
  cookie: { secure: false }
}))

app.set('views', path.join(__dirname, '\\public\\views'))
app.set('view engine', 'ejs')

const newsRouter = require('./routes/news')
const registerRouter = require('./routes/register')
const loginRouter = require('./routes/login')
const homelssueRouter = require('./routes/homelssue')
const engineerRouter = require('./routes/engineer')

app.get('/', (req, res) => {
```

```

    if (req.session.user) {
      res.render('introduction', { login: true })
    } else {
      res.render('introduction', { login: false })
    }
  })

app.get('/introduction', (req, res) => {
  if (req.session.user) {
    res.render('introduction', { login: true })
  } else {
    res.render('introduction', { login: false })
  }
})

app.use('/engineers', engineerRouter)
app.use('/homeIssues', homeIssueRouter)
app.use('/news', newsRouter)
app.use('/register', registerRouter)
app.use('/login', loginRouter)

app.listen(port, (req, res) => {
  console.log(`Server Address http://localhost:${port}`)
  console.log('Click Ctrl + C to quit')
})

```

Login.js

```

const util = require('util');
const sqlite3 = require('sqlite3').verbose()
const express = require('express')
const bcrypt = require('bcryptjs');
const { log } = require('console');

let router = express.Router()

router.get('/', (req, res) => {
  res.render('login', { login: false })
})

router.post('/checkUser', async (req, res) => {
  let db = new sqlite3.Database('10_3D')

  const dbGet = util.promisify(db.get).bind(db)
  let { name, password } = req.body

  try {
    let result = await dbGet('SELECT * FROM User WHERE name = ?', [name])
    if (!result) {
      return res.status(400).send({ msg: 'User not found' })
    }
  }
}

```

```

    }

    let isMatch = await bcrypt.compare(password, result.password)
    if (!isMatch) {
        return res.status(400).send({ msg: 'Invalid password' })
    }

    req.session.user = {
        id: result.id,
        name: result.name
    }

    res.send({
        id: result.id,
        name: result.name
    })
} catch (err) {
    console.error('Database error:', err.message);
    res.status(500).send({ msg: 'Database error' });
} finally {
    db.close()
}
})

module.exports = router

```

Register.js

```

const util = require('util');
const express = require('express');
const sqlite3 = require('sqlite3').verbose()
const bcrypt = require('bcryptjs')

let router = express.Router();

router.get('/', (req, res) => {
    res.render('register', { login: false })
})

router.post('/addUser', async (req, res, next) => {
    let db = new sqlite3.Database('10_3D')

    const dbAll = util.promisify(db.all).bind(db);
    const dbRun = util.promisify(db.run).bind(db);

    const body = req.body;

    if (checkParam(body)) {
        return res.send({ msg: 'data check fail' });
    }
}

```



```

    }

    try {
      const result = await dbAll('SELECT * FROM User WHERE name = ?', [body.name]);

      if (result.length > 0) {
        return res.status(400).send({ msg: 'User already exists' });
      }

      let encrypPwd = await bcrypt.hash(body.password, 10)

      await dbRun('INSERT INTO User(name, password, email, phone, state, subcode)
VALUES(?,?,?,?,?,?)',
        [body.name, encrypPwd, body.email, body.phone, body.state, body.subcode]);

      res.send({msg: 'add user success'});

    } catch (err) {
      console.error('Database error:', err.message);
      res.status(500).send({ msg: 'Database error' });
    } finally {
      db.close((err) => {
        if (err) {
          return console.error(err.message);
        }
        console.log('Closed the database connection.');
```