To **integrate the Frontend Design System** with your **Laravel 11 backend API (from 7211640b-98a4-4e13-8f98-abaf187970b3.md)**, here is a unified **design + API integration approach** for your React frontend. This will help you use the design system components effectively with real backend endpoints.

**🔧 Integration Strategy**

We'll map reusable UI components from the design system to specific API functions from your Laravel backend.

**1. 🔐 Authentication Integration**

**✅ Component: <LoginForm />**

**Uses:**

* POST /api/login

const handleLogin = async (email: string, password: string) => {

try {

const response = await axios.post('/api/login', { email, password });

localStorage.setItem('token', response.data.token);

// Redirect or update context

} catch (error) {

setError('Invalid login credentials.');

}

};

**UI Integration:**

<InputField label="Email" type="email" />

<InputField label="Password" type="password" />

<Button variant="primary" onClick={handleLogin}>Login</Button>

**2. 🧑 User Management (Admin)**

**✅ Component: <UserTable />, <UserForm />**

**APIs:**

* GET /api/users
* POST /api/users
* PUT /api/users/{id}
* DELETE /api/users/{id}

**UI Integration:**

<Button variant="primary" onClick={openCreateUserModal}>Add User</Button>

<Card>

{users.map(user => (

<div>{user.name} <Button onClick={() => deleteUser(user.id)}>Delete</Button></div>

))}

</Card>

**3. 📚 Courses**

**✅ Component: <CourseCard />, <CourseForm />**

**APIs:**

* GET /api/courses
* POST /api/courses
* PUT /api/courses/{id}
* DELETE /api/courses/{id}

**UI Integration:**

<Card>

<h2>{course.title}</h2>

<p>{course.description}</p>

<Button onClick={() => enroll(course.id)}>Enroll</Button>

</Card>

**4. 📦 Lesson Content**

**✅ Component: <LessonCard />, <LessonEditor />**

**APIs:**

* GET /api/lessons
* POST /api/lessons
* GET /api/lessons/{id}

**UI Integration:**

<Card>

<h3>{lesson.title}</h3>

<div>{lesson.content}</div>

</Card>

**5. 📄 Quizzes**

**✅ Component: <Quiz />, <QuestionList />**

**APIs:**

* GET /quiz-questions
* POST /quiz-questions
* GET /api/quizzes

**UI Integration:**

<Quiz>

<Question question="What is React?" />

<InputField type="text" />

</Quiz>

**6. 🧾 Transactions**

**✅ Component: <TransactionTable />**

**APIs:**

* GET /api/transactions
* POST /api/transactions

**UI Integration:**

<Card>

{transactions.map(tx => (

<div>{tx.amount} - {tx.status}</div>

))}

</Card>

**7. 🧪 Progress Tracking**

**✅ Component: <ProgressBar />**

**APIs:**

* GET /api/progress
* POST /api/progress

**UI Integration:**

<ProgressBar value={completedLessons / totalLessons \* 100} />

**8. 🎓 Certificates**

**✅ Component: <CertificateCard />**

**APIs:**

* GET /api/certificates

**UI Integration:**

<Card>

<h4>Certificate for {course.title}</h4>

<a href={certificate.path} target="\_blank">Download PDF</a>

</Card>

**9. 🔔 Notifications**

**✅ Component: <NotificationList />**

**APIs:**

* GET /notifications
* POST /notifications

**UI Integration:**

<Alert variant="info">You’ve enrolled in Laravel Basics!</Alert>

**🧩 Example File Structure (React + API Integration)**

src/

├── components/

│ ├── Button.tsx

│ ├── Card.tsx

│ ├── InputField.tsx

│ ├── Navbar.tsx

├── pages/

│ ├── LoginPage.tsx

│ ├── CoursesPage.tsx

│ ├── DashboardPage.tsx

├── services/

│ ├── authService.ts

│ ├── userService.ts

│ ├── courseService.ts

├── styles/

│ └── design-system.scss

Here's a **Frontend Design System** tailored for your **React-based Online Course Platform**, ensuring a consistent, reusable, and accessible UI — all based on your brand colors (#6151C6, #D9D9D9, #000000, #FFFFFF).

**🎨 Frontend Design System**

**1. 🎨 Color Tokens**

| **Token** | **Value** | **Usage** |
| --- | --- | --- |
| --color-primary | #6151C6 | Buttons, links, accents |
| --color-secondary | #D9D9D9 | Backgrounds, subtle elements |
| --color-dark | #000000 | Headings, body text |
| --color-light | #FFFFFF | Cards, page backgrounds |

:root {

--color-primary: #6151C6;

--color-secondary: #D9D9D9;

--color-dark: #000000;

--color-light: #FFFFFF;

}

**2. 🖋️ Typography**

| **Element** | **Font Size** | **Font Weight** | **Line Height** | **Color** |
| --- | --- | --- | --- | --- |
| Heading 1 | 32px | 700 | 1.25 | #000000 |
| Heading 2 | 24px | 600 | 1.3 | #000000 |
| Body Text | 16px | 400 | 1.6 | #000000 |
| Caption/Text | 14px | 300 | 1.6 | #6151C6 (e.g. muted links) |

**Font Family:** 'Segoe UI', sans-serif

**3. 📦 Components**

**✅ Buttons**

| **Type** | **Class** | **Color** |
| --- | --- | --- |
| Primary | .btn-primary | --color-primary, white text |
| Secondary | .btn-secondary | --color-secondary, dark text |

.btn-primary {

background-color: var(--color-primary);

color: var(--color-light);

border: none;

padding: 0.6rem 1.2rem;

border-radius: 6px;

transition: background 0.3s;

}

.btn-primary:hover {

background-color: darken(var(--color-primary), 10%);

}

**🧱 Cards**

.card {

background-color: var(--color-light);

border: 1px solid var(--color-secondary);

padding: 1rem;

border-radius: 10px;

box-shadow: 0 2px 4px rgba(0,0,0,0.05);

}

**🗂️ Form Fields**

* Full-width inputs
* Rounded edges
* Clear labels above inputs
* Example:

.input-field {

border: 1px solid var(--color-secondary);

border-radius: 8px;

padding: 0.5rem 1rem;

width: 100%;

font-size: 16px;

color: var(--color-dark);

}

.input-field:focus {

border-color: var(--color-primary);

outline: none;

}

**4. 🔄 Layout System**

**Breakpoints**

| **Device** | **Width** |
| --- | --- |
| Mobile | < 576px |
| Tablet | ≥ 576px |
| Desktop | ≥ 992px |

Use Bootstrap Grid or custom Flexbox for layout.

**Navbar**

* Fixed top
* Includes logo, navigation links, profile/logout
* Mobile menu collapses into hamburger

**5. 🌘 Theme Modes (optional)**

Add dark mode toggle:

body.dark {

--color-light: #1a1a1a;

--color-dark: #f1f1f1;

--color-secondary: #333;

}

Toggle with a button stored in localStorage.

**6. 🧩 Reusable React Components**

| **Component** | **Description** |
| --- | --- |
| <Button /> | Accepts props: variant, onClick, children |
| <Card /> | Used for course, quiz, and user display |
| <InputField /> | Label + input, reusable in forms |
| <Navbar /> | Site-wide navigation |
| <Sidebar /> | Admin dashboard (optional) |
| <CourseCard /> | Course listing |
| <Quiz /> | Quiz question UI |

**7. ⚠️ Feedback & Alerts**

* Use Bootstrap’s alerts or create:

.alert {

border-radius: 6px;

padding: 0.75rem 1.25rem;

margin-bottom: 1rem;

}

.alert-success {

background-color: #d1e7dd;

color: #0f5132;

}

.alert-error {

background-color: #f8d7da;

color: #842029;

}

**8. 🧪 Example: Reusable Button Component (React)**

type Props = {

variant?: 'primary' | 'secondary';

onClick?: () => void;

children: React.ReactNode;

};

export const Button = ({ variant = 'primary', onClick, children }: Props) => (

<button className={`btn btn-${variant}`} onClick={onClick}>

{children}

</button>

);