

I'll provide the necessary frontend functionality using Angular and PrimeNG to integrate with the backend APIs for quiz management.

---

## 1. Install Required Packages

Run these commands in your Angular project:

```
npm install primeng primeicons  
npm install @angular/forms @angular/common @angular/router
```

Update `angular.json` to include PrimeNG styles:

```
"styles": [  
  "node_modules/primeng/resources/themes/lara-light-blue/theme.css",  
  "node_modules/primeng/resources/primeng.min.css",  
  "node_modules/primeicons/primeicons.css",  
  "src/styles.css"  
]
```

---

## 2. Create Quiz Management Module

Run:

```
ng generate module quiz  
ng generate component quiz/create  
ng generate component quiz/take  
ng generate component quiz/result
```

---

## 3. Define Quiz Model (`quiz.model.ts`)

```
export interface Quiz {  
  quizId?: number;  
  quizName: string;  
  courseId: number;  
  description: string;  
  totalMarks: number;  
  createdBy: number;  
}
```

```

export interface Question {
  questionId?: number;
  quizId: number;
  questionText: string;
  correctOptionId: number;
}

export interface Option {
  optionId?: number;
  questionId: number;
  optionText: string;
}

export interface QuizAttempt {
  attemptId?: number;
  userId: number;
  quizId: number;
  score?: number;
  attemptDate?: Date;
}

export interface Answer {
  answerId?: number;
  attemptId: number;
  questionId: number;
  selectedOptionId: number;
}

```

---

## 4. Create Quiz Service (**quiz.service.ts**)

```

import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { Observable } from 'rxjs';
import { Quiz, Question, QuizAttempt, Answer } from './quiz.model';

@Injectable({
  providedIn: 'root'
})
export class QuizService {
  private apiUrl = 'http://localhost:3000/quiz';

  constructor(private http: HttpClient) {}

  createQuiz(quiz: Quiz): Observable<Quiz> {
    return this.http.post<Quiz>(`${this.apiUrl}/create`, quiz);
  }

```

```

    }

    getQuizzes(): Observable<Quiz[]> {
        return this.http.get<Quiz[]>(`${this.apiUrl}`);
    }

    getQuestions(quizId: number): Observable<Question[]> {
        return this.http.get<Question[]>(`${this.apiUrl}/${quizId}/questions`);
    }

    submitQuizAttempt(attempt: QuizAttempt): Observable<QuizAttempt> {
        return this.http.post<QuizAttempt>(`${this.apiUrl}/submit`, attempt);
    }
}

```

---

## 5. Implement Tutor's Quiz Creation UI (**create-quiz.component.ts**)

```

import { Component } from '@angular/core';
import { QuizService } from '../quiz.service';
import { Quiz } from '../quiz.model';

@Component({
    selector: 'app-create-quiz',
    templateUrl: './create-quiz.component.html'
})
export class CreateQuizComponent {
    quiz: Quiz = {
        quizName: "",
        courseId: 1, // Set dynamically
        description: "",
        totalMarks: 100,
        createdBy: 1 // Tutor's ID
    };

    constructor(private quizService: QuizService) {}

    createQuiz() {
        this.quizService.createQuiz(this.quiz).subscribe(response => {
            alert('Quiz Created Successfully');
        });
    }
}

```

## Create Quiz UI (**create-quiz.component.html**)

```
<p-card header="Create Quiz">
  <div>
    <label>Quiz Name:</label>
    <input type="text" [(ngModel)]="quiz.quizName" class="p-inputtext p-mb-2" />
  </div>

  <div>
    <label>Description:</label>
    <textarea [(ngModel)]="quiz.description" class="p-inputtext p-mb-2"></textarea>
  </div>

  <button pButton type="button" label="Create Quiz" (click)="createQuiz()"></button>
</p-card>
```

---

## 6. Implement Quiz Taking UI (**take-quiz.component.ts**)

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute } from '@angular/router';
import { QuizService } from '../quiz.service';
import { Question, Answer, QuizAttempt } from '../quiz.model';

@Component({
  selector: 'app-take-quiz',
  templateUrl: './take-quiz.component.html'
})
export class TakeQuizComponent implements OnInit {
  quizId!: number;
  userId = 2; // Example student ID
  questions: Question[] = [];
  answers: Answer[] = [];

  constructor(private quizService: QuizService, private route: ActivatedRoute) {}

  ngOnInit() {
    this.quizId = Number(this.route.snapshot.paramMap.get('quizId'));
    this.quizService.getQuestions(this.quizId).subscribe(data => {
      this.questions = data;
      this.answers = this.questions.map(q => ({
        attemptId: 0,
        questionId: q.questionId,
        selectedOptionId: 0
      }));
    });
  }
}
```

```

    });
  }

  submitQuiz() {
    const attempt: QuizAttempt = {
      userId: this.userId,
      quizId: this.quizId
    };

    this.quizService.submitQuizAttempt(attempt).subscribe(result => {
      alert(`Quiz Submitted. Score: ${result.score}`);
    });
  }
}

```

## Take Quiz UI (**take-quiz.component.html**)

```

<p-card header="Take Quiz">
  <div *ngFor="let question of questions">
    <p>{{ question.questionText }}</p>
    <p-radioButton *ngFor="let option of question.options"
      name="question{{question.questionId}}"
      [(ngModel)]="answers[question.questionId].selectedOptionId"
      [value]="option.optionId">
      {{ option.optionText }}
    </p-radioButton>
  </div>

  <button pButton type="button" label="Submit Quiz" (click)="submitQuiz()"></button>
</p-card>

```

---

## 7. Implement Results Page (**result.component.ts**)

```

import { Component, OnInit } from '@angular/core';
import { QuizService } from '../quiz.service';
import { QuizAttempt } from '../quiz.model';

@Component({
  selector: 'app-result',
  templateUrl: './result.component.html'
})
export class ResultComponent implements OnInit {
  userId = 2; // Student ID
  results: QuizAttempt[] = [];

```

```

constructor(private quizService: QuizService) {}

ngOnInit() {
  this.quizService.getResults(this.userId).subscribe(data => {
    this.results = data;
  });
}
}

```

## Results UI (**result.component.html**)

```

<p-card header="Quiz Results">
  <p-table [value]="results">
    <ng-template pTemplate="header">
      <tr>
        <th>Quiz ID</th>
        <th>Score</th>
        <th>Attempt Date</th>
      </tr>
    </ng-template>
    <ng-template pTemplate="body" let-result>
      <tr>
        <td>{{ result.quizId }}</td>
        <td>{{ result.score }}</td>
        <td>{{ result.attemptDate }}</td>
      </tr>
    </ng-template>
  </p-table>
</p-card>

```

---

## 8. Routing Configuration (**app-routing.module.ts**)

```

const routes: Routes = [
  { path: 'create-quiz', component: CreateQuizComponent },
  { path: 'take-quiz/:quizId', component: TakeQuizComponent },
  { path: 'results', component: ResultComponent },
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {}

```

---

## Final Steps

- **Tutor Dashboard:** List quizzes and manage them.
- **Student Dashboard:** Display available quizzes and allow taking quizzes.
- **Payment Integration:** Add Razorpay/Stripe for payments.
- **Styling:** Use PrimeNG `p-card`, `p-button`, and `p-table` for UI.

Let me know if you need additional functionality! 🚀