[Create Quiz]

Technical Document

Date: 13/Feb/2019

Version: Version 0.1

# Table of Contents

[Table of Contents 1](#_Toc1073990)

[[Task Name] 2](#_Toc1073991)

[Description 2](#_Toc1073992)

[1.1 Mockup screen: 2](#_Toc1073993)

[1.2 Angular & Spring Models 3](#_Toc1073994)

[1.3 Angular Component 4](#_Toc1073995)

[1.4 Rest Services 4](#_Toc1073996)

[1.5 Flow Services: 4](#_Toc1073997)

[1.6 Repository Functions 5](#_Toc1073998)

[1.7 Database Script: 5](#_Toc1073999)

[1.8 ER Diagram: 7](#_Toc1074000)

# 

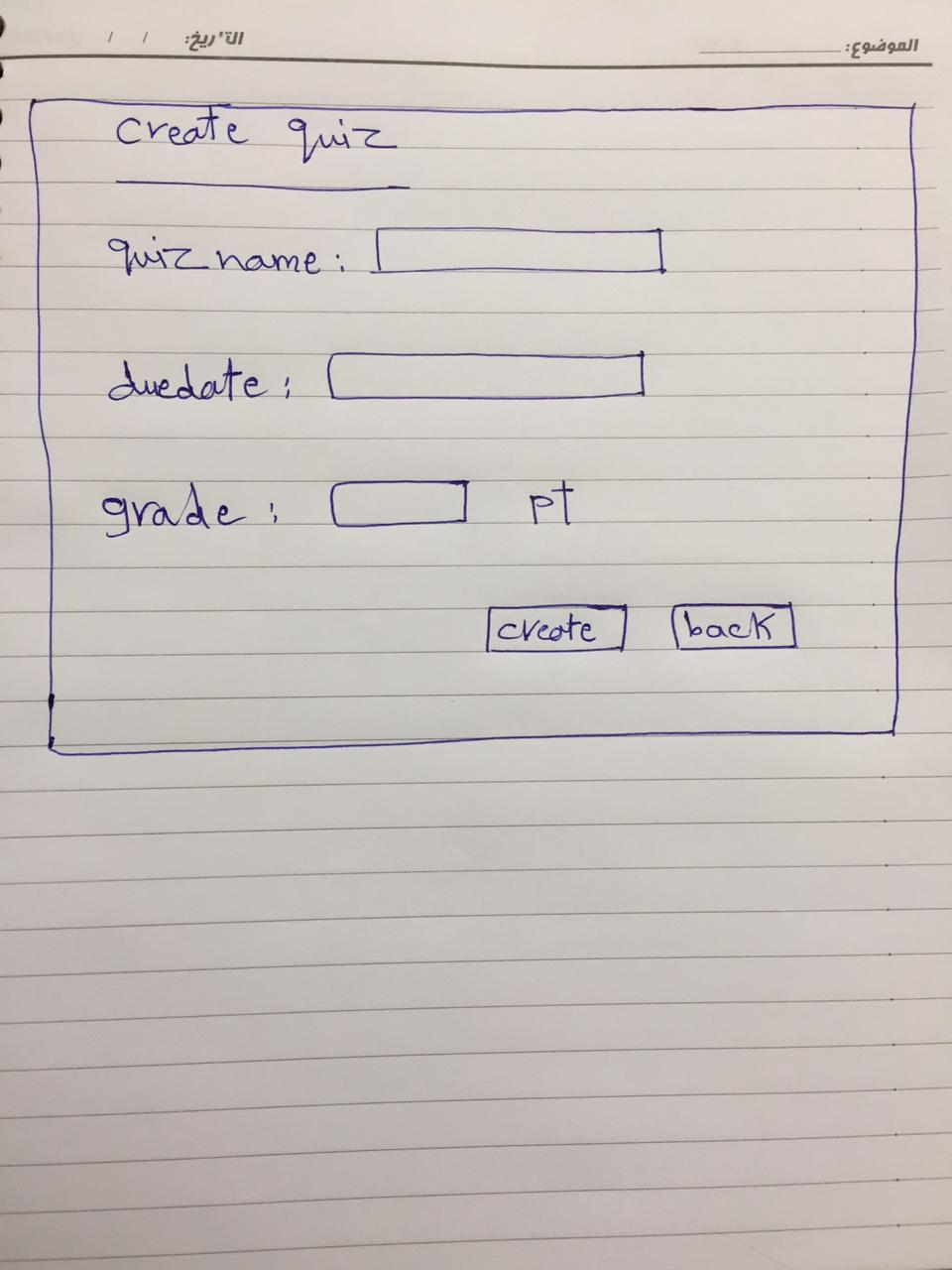
# [create quiz]

Refer to: [Taiga URL]

## Description

Create quiz in the course .

## Mockup screen:



## Angular & Spring Models

|  |  |  |  |
| --- | --- | --- | --- |
| **New** | **Class Name** | QuizDTO | |
| **Angular Path** | | app/modules/course/shared/data/quiz/quiz-dto.ts | |
| **Spring Path** | | model/course/QuizDTO | |
| **Fields** | | | |
| **Attribute** | | **Angular Data Type** | **Spring Data Type** |
| quizName | | string | String |
| grade | | number | int |
| dueDate | | DATE | Date |

## Angular Component

|  |  |  |
| --- | --- | --- |
| **New** | **Class Name** | CreateQuiz |
| **Path** | | app/modules/course/components/quiz/create-quiz.component.[ts/html] |
|  | |  |
| **new** | **Class Name** | QuizService |
| **Path** | | app/modules/course/shared/quiz.service.ts |
| **Function** | | createNewQuiz(quiz: QuizDTO) |

## Rest Services

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **New** | **Name** | | api/course/{courseID}/quiz**/** | | | **Method** | POST |
| **Controller** | | | CourseRes | | | **Function** | createQuiz(int courseID,QuizDTO) |
| **REST Parameter** | | | | | | | |
| **Attribute** | | **Parameter Type** | | **Data Type** | **Description** | | |
| quiz | | Payload | | QuizDTO |  | | |

## Flow Services:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **New** | **Class Name** | CourseSer | **Function** | createQuiz(int courseID ,QuizDTO quiz) |
| **Description** | | Add New quiz | | |
| **Steps** | | | | |
| 1. Check if the user is Instructor 2. Validate quiz Data 3. Insert quiz Data in Database 4. Return the Quiz ID after creation (Yara can help you with that) | | | | |

## Repository Functions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **New** | **Class Name** | QuizRep | **Type** | Insert SQL |
| **Function** | | insertNewQuiz(int courseID ,QuizDTO quiz) | | |
| **Query** | | | | |
| INSERT INTO quiz (quiz\_name, due\_date,grade ,course\_id ) VALUES (?, ?, ?,?) | | | | |

## Database Script:

**CREATE TABLE** quiz (  
 **id INT NOT NULL** AUTO\_INCREMENT,  
 **quiz\_name VARCHAR**(45) **NOT NULL**,  
 **grade INT NOT NULL**,  
 **date DATE NOT NULL**,  
 **course\_id INT NOT NULL**,  
 **PRIMARY KEY** (**id**),  
 **CONSTRAINT** fk\_quiz\_course  
 **FOREIGN KEY** (**course\_id**)  
 **REFERENCES** course (id v)  
 **ON DELETE NO ACTION  
 ON UPDATE NO ACTION**);

**ALTER TABLE** quiz  
CHANGE **COLUMN date** due\_date **DATE NOT NULL** ;

## ER Diagram:

