

Project specification

- The product backlog: a complete list of all functionality (i.e., the actions) of your project, and an English description of each action. We strongly suggest that you organize these features into groups/modules based on related functionality.

1. Quiz creator
 - quiz pool
 - basic function(set up questions, answers, score weights)
 - management function(available time)
 - generate PIN code
 - categories: Multiple choice
2. Quiz access
 - PIN access
 - fill up the quiz and submit
 - see their final score/answers
3. Quiz result(to the creator)
 - Answers, the total score of each participant
 - statistics(distribution, average score, lowest/highest score)
 - output to Excel format
4. Leaderboard
5. Login/Register
6. Basic HTML pages

- The first sprint backlog: a complete list of the functionality you will complete during your first sprint, and how that work is allocated among your team members.

1. Login/Register
2. Basic HTML pages
3. Quiz Creation
 - a. basic functions (set up questions, answers, score weights)
 - b. question category: multiple-choice

Mingyue Li : 3.Quiz Creation

Jintao Huang: 3.Quiz Creation

Yiqin Han: 1, 2

- The name and Andrew ID of the product owner for the first sprint.
Yiqin Han, yiqinh

- A complete implementation of the data models used by your application. This may be written in as Django models, SQL, or some equivalent style implementation if you use another framework.

We have five objects to be defined in our database, person, quizlet, question and options.

The quizlet and questions are many-to-many relationships. A quizlet can have many questions, and the questions could belong to multiple quizlet(Quiz pool, you can apply the same question for multiple quizlet).

The options and questions are a many-to-one relationship. One question can have multiple options but one option can only belong to one question.

The person provides answers to questions. And he only can choose one option for the question.

A copy of the current models.py

```
import datetime
from typing import Any

from django.db import models
from django.contrib.auth.models import User

class Quiz(models.Model):
    title = models.CharField(max_length=100)
    description = models.CharField(max_length=400)
    start_date = models.DateTimeField('date published')
    end_date = models.DateTimeField('date ended')

    def __str__(self):
        return self.title

class Question(models.Model):
    question_text = models.CharField(max_length=200)
    score = models.IntegerField()
    quiz = models.ForeignKey(Quiz, on_delete=models.PROTECT,
related_name="question")

    def __str__(self):
        return self.question_text

class Choice(models.Model):
    choice_text = models.CharField(max_length=200)
    is_answer = models.BooleanField()
    quiz = models.ForeignKey(Quiz, on_delete=models.PROTECT,
```

```

related_name="choice")
    question = models.ForeignKey(Question, on_delete=models.PROTECT,
related_name="choice")
    # user = models.ManyToManyField(User)

    def __str__(self):
        return self.choice_text

# class Quiz_Question_Choice(models.Model):
#     quiz = models.ForeignKey(Quiz)
#     question = models.ForeignKey(Question)
#     Choice = models.ForeignKey(Choice)

class Person_Answer(models.Model):
    quiz = models.ForeignKey(Quiz, on_delete=models.PROTECT,
related_name="person_answer")
    question = models.ForeignKey(Question, on_delete=models.PROTECT,
related_name="person_answer")
    choice = models.ForeignKey(Choice, on_delete=models.PROTECT,
related_name="person_answer")
    user = models.ForeignKey(User, on_delete=models.PROTECT,
related_name="person_answer")

```

- A complete set of drawn wireframes or HTML mockups for your application, for all non-trivial views within the application.

The welcome page:



The feature page:

Features



Create Quizzes

Select the appropriate question type(s), write your question(s)/answer(s) and assign score for each question

- ✓ Multiple question types
- ✓ Easy to manage
- ✓ Auto grader
- ✓ Quiz pool for random questions



Complete Quizzes

Access quizzes simply through a code! Provide you answers, submit it and your result will be shown

- ✓ Access through PIN code
- ✓ View your answers and score
- ✓ Answers will be stored and recorded for future review



Analyze Results

Quiz creators can view the results at anytime!

- ✓ View each individual's answers
- ✓ View statistics about the quizzes
- ✓ Output the data to Excel form

Function page:

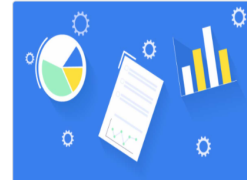
Let's begin!



Create a quiz



Take a quiz



Quiz history