

Project Specification

Project Proposal

This is an online K-12 & pre-K education website. The website aims to provide students and their parents with a comprehensive platform to study, test and communicate.

The website consists of several parts: test and learn, fun with learning and forums. Students can learn new knowledge by taking online tests or exams. Students can play interesting games to strengthen their previously learned knowledge with fun.

It is well noted that the social media function of this project will not take an important portion. Therefore, the discussion forum and communication function between parents/students will not be prioritized in this project specification.

Database Models

We use django framework to develop the whole project. The database model mainly consists of four parts: user and their profiles, scoreboard and user statistics.

Each user has its own profile(including bio, avatar and friends info), total score and tier-ladder ranking scores.

Each user has its recent activity stats for each exams or tests and their scores.

Each question in question pool has its content, choices and correct answer.

The following code snippets demonstrate the django database model of this project.

```
from django.db import models

# Create your models here.

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

class Category(models.Model):
    name = models.TextField(default="", max_length=50)
```

```

# User's recent activity or exam/battle scores
class UserStats(models.Model):
    # If the score is smaller than 0, then it represent learning activity
    score = models.IntegerField(default=0)
    datetime = models.DateTimeField(auto_now=True)
    categories = models.ManyToManyField(Category, related_name="stats_category")

# Content of question pool
class Questions(models.Model):
    question = models.TextField(default="", max_length=200)
    # The format of choice shall be JSON
    choice = models.TextField(default="", max_length=1000)
    correct_answer = models.IntegerField(default=0)
    # Indicates categories of questions (e.g, English, Maths..)
    categories = models.ManyToManyField(Category, related_name="question_category")

    def __str__(self):
        return self.question

class Profile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    bio = models.TextField(default="", max_length=300, blank=True)
    avatar = models.ImageField(default="avatar/no_image.jpg", upload_to='avatar/',
    blank=True)
    # K-12 Grade, 0 for kindergarten, 1 for first grade, 9 for 9th grade and so on
    grade = models.IntegerField(default=0)
    friends = models.ManyToManyField(User, related_name="friends", blank=True)

    # Information on global scoreboard, display in user's profile is optional.
    score = models.IntegerField(default=0)
    level = models.IntegerField(default=0)

    # Stats for each user
    stats = models.ManyToManyField(UserStats, related_name="user_stats", blank=True
)

    def __str__(self):
        return self.bio

```

List of Functionality

User authentication and profile settings

New users are able to sign up with their emails and basic information. They shall select grades before they proceed using all the features on the learning platform. Users can set their profile

bio and upload profile images on profile page. Users can connect with friends by searching or automatic matching for the sake of competing in exams.

Question Pool and their categories

Users can learn, take exams and play game in the range of following categories:

- Mathematics
- English
- Geography

Settings

Level: Where you can set and modify your current level - from first grade in primary school to the middle school.

Profile: Setting and modifying your profile.

Fun with learning

Memorizing Game:

There is a number of blocks that contains pictures or words. In the first 10 seconds, player will see the relationship between pictures and words. In the next 5 seconds, player will see the shuffled blocks of pictures and words. Then, the blocks will be reversed. Players need to choose the matching pair of blocks.

If players correctly chose a pair, then the pair will stay turned. If players do not choose a correct pair, the pair will be back to reverse status.

Counting hamburgers:

This is a simple game for kindergarten kids to count numbers.

Given four plates of hamburgers, player should choose the right amount of hamburgers in each plate. The player has total limited time to play this game. The final result is the number of questions player complete.

Sudoku:

For middle school students, they can choose to solve sudoku. The objective is to fill a 9×9 grid with digits so that each column, each row, and each of the nine 3×3 subgrids that compose

the grid contains all of the digits from 1 to 9.

The number of sudoku puzzle is limited, not dynamically generated.

Test and Learn

The questions of the following sections shall fit in previously mentioned categories and randomly selected by system.

Battle in tier-ladder:

Each player starts at level 1, after they win a contest with another child, he will get a rise in the level. The contest is about answering questions and give scores based on the correctness as well as the token time. One that win 3 contests continually can rise 2 level, while losing a game can cause a drop in level by one.

Battle with friends:

Player can invite one of his friends to join the two-player contest. Player and his friend have limited time to answer questions. Each correct answer will credit the player with points. The player with highest point wins. The result of the battle will be saved to UserStats.