



OnTrack's CSE Grader Assignment Rails Application

Jing George, Jack Hanley, Von Mbah, Jack Thompson, Snigdha Tiwari



PROJECT GOAL

01

INITIAL APPROACH

02

CORE FEATURES AND
DESIGN

03

DEMO

04

DATABASE SCHEMA

05

TABLE OF CONTENTS

06

OBSTACLES



07

TACKLING
OBSTACLES

08

WHAT WE'VE
LEARNED

09

IMPROVEMENTS

10

EXTRA

The background is a light gray with abstract geometric patterns. On the left, there is a dense network of thin gray lines connecting small dark gray dots, forming a complex web. Scattered across the background are several thin-lined triangles of various sizes and orientations. In the upper right corner, there are small, faint circles or dots. The text "Project Goal" is centered in a large, bold, dark gray font.

Project Goal



Initial Approach


1. Planning
 - a. Basic Idea
 - b. User Requirements
 2. Design
 - a. Database
 - b. Web Application
- 



Core Features

- Available Classes
 - Student Applications
 - Recommendation Submission
 - Administrator Panel
-

Extra Features

- Account System
 - Evaluation
 - Configuration
- 

Core Features

- Available Classes
- Student Application
- Recommendation Form
- Administrator Panel

[Edit Profile](#) [Logout](#)

New Form

[Exit](#)

Academic Info

Courses Completed

Course	Grade	
CSE 1110	A	Delete this Class

[Add a Course](#)

Time Available

Course	
CSE 1110	<input type="checkbox"/> WF 12:40 - 13:35 <input type="checkbox"/> M 12:40 - 13:35

Classes Graded In the Past

Course	
CSE 2231	Delete this Class

[Add a Graded Course](#)

Administrator Panel

[All semesters](#) [All courses](#) [All sessions](#) [Courses with and without open positions](#) [Filter](#)

CSE 1110 Introduction to Computing Technology U 3, Class #7706

[Edit](#)

Semester: SP20

Session: Regular

Component: LEC

0 of 0 grading positions open. Graders assigned:

Location N/A

ARR

Mallon, Michelle; Shen, Jingyi

Extra Features

- Account System
- Evaluation
- Configuration

CSE Grader Assignment System

by OnTrack

Menu

Sign up

Login

Update courses (this will overwrite all course info, including current grader assignments)

Run Web Scraper

View or create a configuration

I want to change the configuration for all classes that the following criteria:

*** Leave a field blank if no preference. ***

*** To list multiple identifiers of one type, separate each by a comma and space. Example: 3901, 3902, 3903 ***

Semester (ex: SU20):

Component (ex: LAB):

Class number (unique):

Number in name (ex: 2221):

Number of graders:

☐ Grader attendance is required

☐ Grader attendance is not required

Update

DEMO



Database Schema

student_recommend

id	int
student_fname	varchar
student_lname	varchar
student_name_num	int
teacher_lname	varchar
teacher_name_num	int
course	varchar
course_section	varchar
email	varchar
semester	varchar
course_specification	varchar

Evulation

id	int
instructor_fname	varchar
instructor_lname	varchar
instructor_name_dotnum	varchar
grader_lname_dotnum	varchar
course	varchar
quality	int
punctuality	int
com_skills	int
course_knowledge	int

Course

id	int
class_num	int
semester	varchar
name	varchar
session	varchar
component	varchar
attendance	varchar
num_graders	varchar

Teacher_User

id	int
email	varchar
encrypted_password	varchar
reset_password_token	varchar
reset_password_sent_at	varchar
remember_created_at	varchar
f_name	varchar
l_name	varchar
dot_num	int

Grader_Completed_Course

id	int
grader_id	int
course_id	int
grade	varchar

Instructor

id	int
name	varchar

Course_Grader

grader_id	int
course_id	int

Class_Name

id	int
name	varchar

Admin

id	int
email	varchar
encrypted_password	varchar
reset_password_token	varchar
reset_password_sent_at	varchar
remember_created_at	varchar

Grader_Previous_Grade_Course

id	int
grader_id	int
course_id	int

Instructor_Meeting

meeting_id	int
instructor_id	int

Grader

id	int
name	varchar
last_name_dot	varchar
gpa	decimal

Teaching

id	int
class_number	int
component	varchar
location	varchar
instructor	varchar
session	varchar
topic	varchar
semester	varchar
class_name_id	int

Grader_Time_Availability

id	int
grader_completed_course_id	int
time	varchar

Meeting

id	int
location	varchar
time	varchar
course_id	int

User

id	int
email	varchar
encrypted_password	varchar
reset_password_token	varchar
reset_password_sent_at	varchar
remember_created_at	varchar
f_name	varchar
l_name	varchar
dot_num	int
gpa	decimal
admin	varchar
category	varchar
profile_id	int

Obstacles

1. Dynamic Forms in Student Applications
 - a. Besides basic information like name and gpa, students need to add a # of classes they completed, and dictate which times they are available.
2. Poor Initial Database Schema
 - a. The initial database schema we came up during Project #3 couldn't be used very well moving into this final project
3. Mismatched Tables
 - a. User table contained both the fields needed for recommendation forms and sign up/login fields which created errors in the recommendation form
4. Working with complicated Rails gems
 - a. Devise



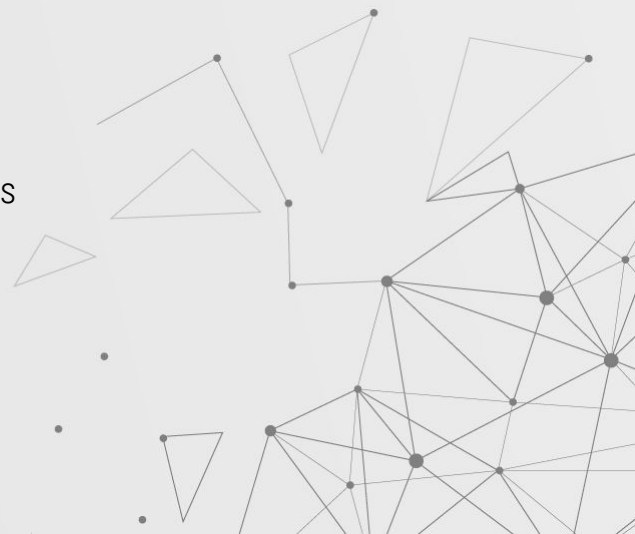
Tackling Obstacles

- Incorporating Javascript and HTTP Request
- Nested Hashes and Arrays in Parameters
- Created new and meaningful tables for the schema
- Separated the columns in User to be in a new table



What We Have Learned

- Coming up with good Database Schemas
 - Initial and good schemas help build a foundation for the project
- Better Documentation
 - All teammates coordinated with each other on the application
- Do Zoom Meetings to Discuss Problems
 - 15 min check-ins to update progress
 - Bouncing bugs off of another person often helped to solve issues
- Importance of Git



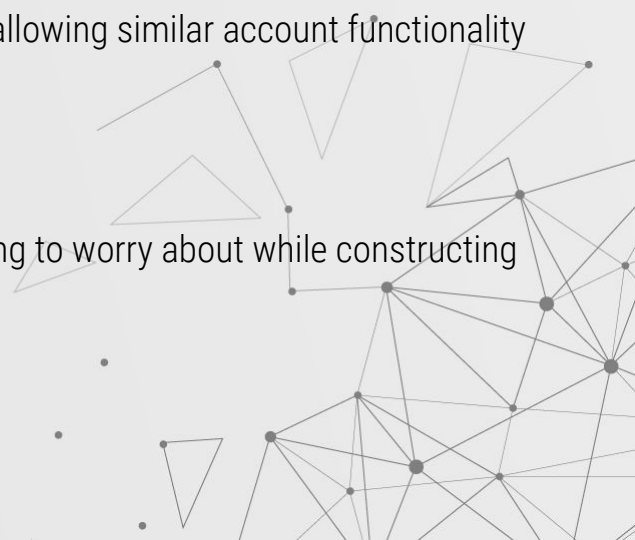
Improvements For V2

- Filtering and sorting the list of candidates eligible to grade a class
- More settings for web scraping
- Error Handling for Edit Recommendations Form
- Cohesive appearance between pages
- Features inspired by other groups' v1



Extra

- We found the use of two gems particularly helpful: Devise and Mechanize
- Devise Advantages:
 - Streamlined the process of implementing user accounts - automatically generates models and views for account creation and updating
 - Built in security features such as hashing passwords
 - Convenient account functionality such as “Remember Me”
 - Could quickly create several similar models for various types of accounts
 - Easier to separate different functions of the application while allowing similar account functionality (Ex. a student could not fill out an evaluation)
- Mechanize Advantages:
 - Made it much simpler to scrape the website and get accurate data
 - Having accurate data from the beginning gave the group one less thing to worry about while constructing the application





THANKS

Does anyone have any questions?

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

Please keep this slide for attribution.

Instructions for use

In order to use this template, you must credit [Slidesgo](#) by keeping the [Thanks](#) slide.

You are allowed to:

- Modify this template.
- Use it for both personal and commercial projects.

You are not allowed to:

- Sublicense, sell or rent any of Slidesgo Content (or a modified version of Slidesgo Content).
- Distribute Slidesgo Content unless it has been expressly authorized by Slidesgo.
- Include Slidesgo Content in an online or offline database or file.
- Offer Slidesgo templates (or modified versions of Slidesgo templates) for download.
- Acquire the copyright of Slidesgo Content.

For more information about editing slides, please read our FAQs or visit Slidesgo School:

<https://slidesgo.com/faqs> and <https://slidesgo.com/slidesgo-school>

Fonts & colors used


This presentation has been made using the following fonts:

Exo 2


(<https://fonts.google.com/specimen/Exo+2>)

Roboto Condensed

(<https://fonts.google.com/specimen/Roboto+Condensed>)

A dark gray rounded square color swatch.

#434343

A light gray rounded square color swatch.

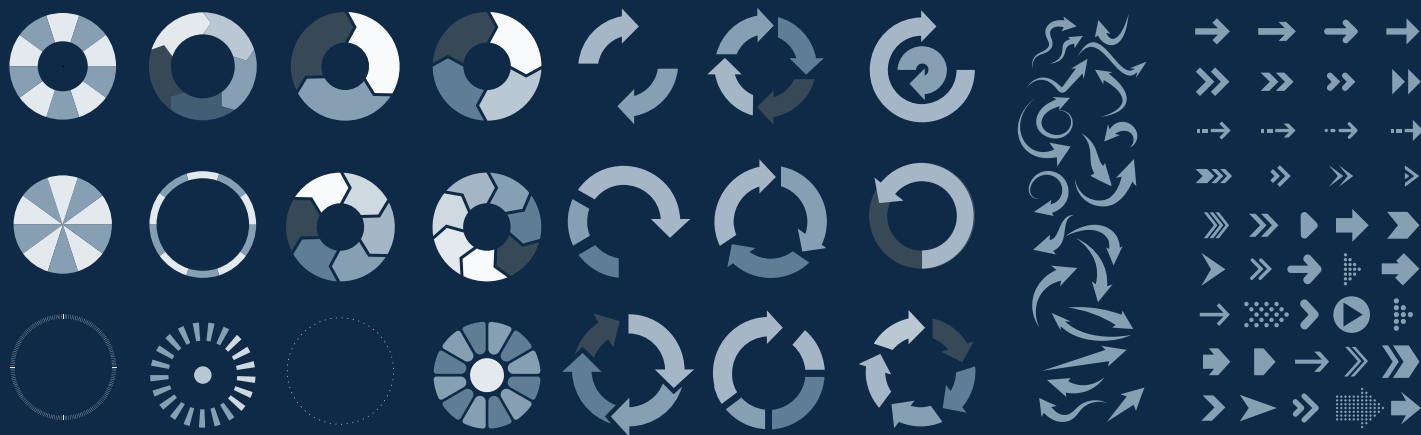
#eeeeee

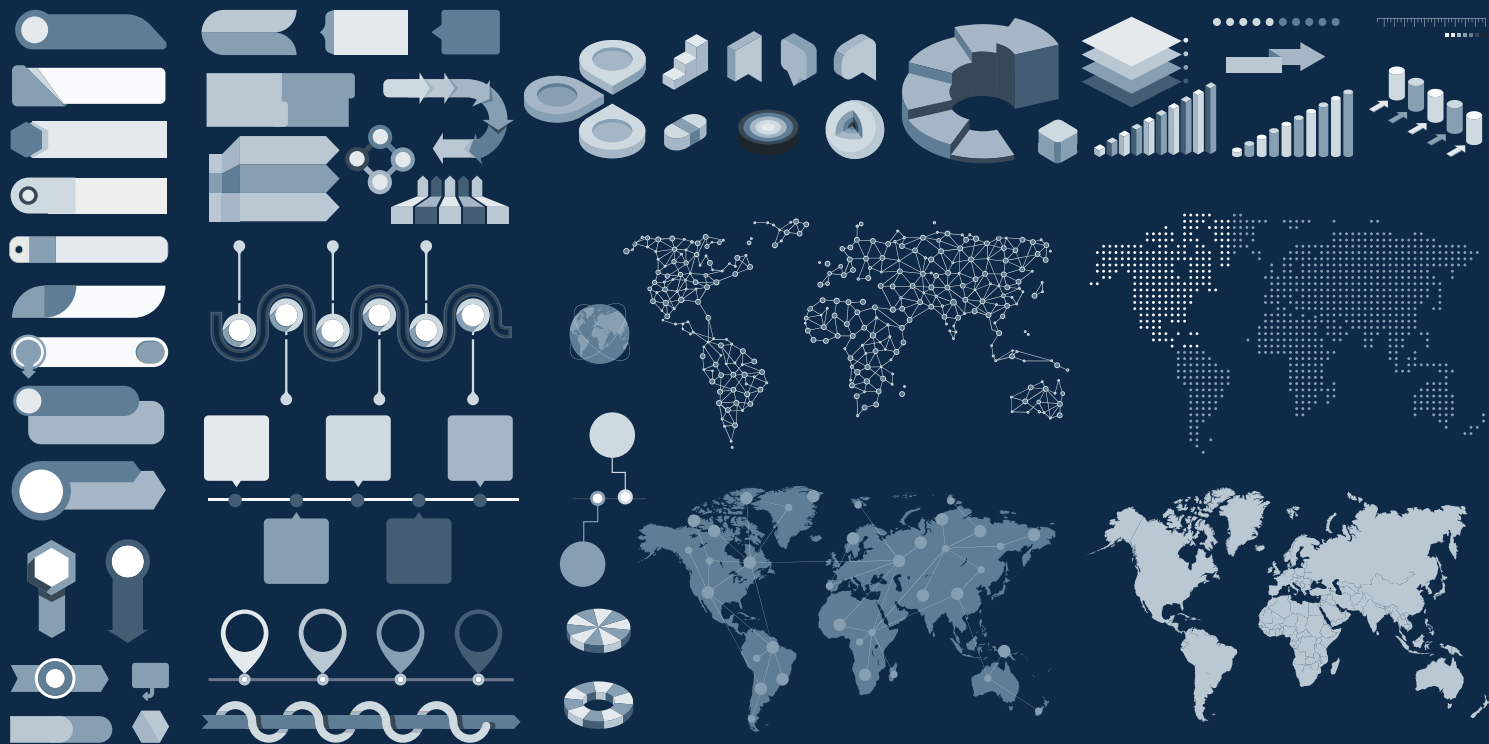
A white rounded square color swatch.

#ffffff

Use our editable graphic resources...

You can easily **resize** these resources, keeping the quality. To **change the color**, just ungroup the resource and click on the object you want to change. Then, click on the paint bucket and select the color you want. Don't forget to group the resource again when you're done.





...and our set of editable icons

You can resize these icons, keeping the quality.

You can change the stroke and fill color; just select the icon and click on the paint bucket/pen.

In Google Slides, you can also use Flaticon's extension, allowing you to customize and add even more icons.



Business Icons



Avatar Icons



Creative Process Icons



Educational Process Icons



Help & Support Icons



Medical Icons



Nature Icons



Performing Arts Icons



SEO & Marketing Icons



Teamwork Icons



