

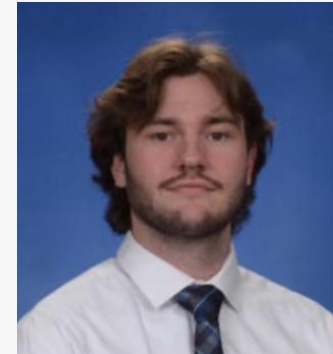
Brady Amundson, CJ Plantemoli,  
and Sam Richard

# Who are we?

Brady Amundson



CJ Plantemoli



Sam Richard



# Welcome to Gruuper

*For Teachers  
& Professors*



# gruuper

# The Problem

Challenges in class  
group formation

Creating groups is  
time-consuming and  
frustrating

Poorly chosen groups  
creates friction and  
limits productivity

The perfect group  
formation tool has  
yet to exist...



# 4 Easy Steps

1

Create  
Account

2

Professor  
creates  
Classroom

3

Students fill  
out profiles  
& join

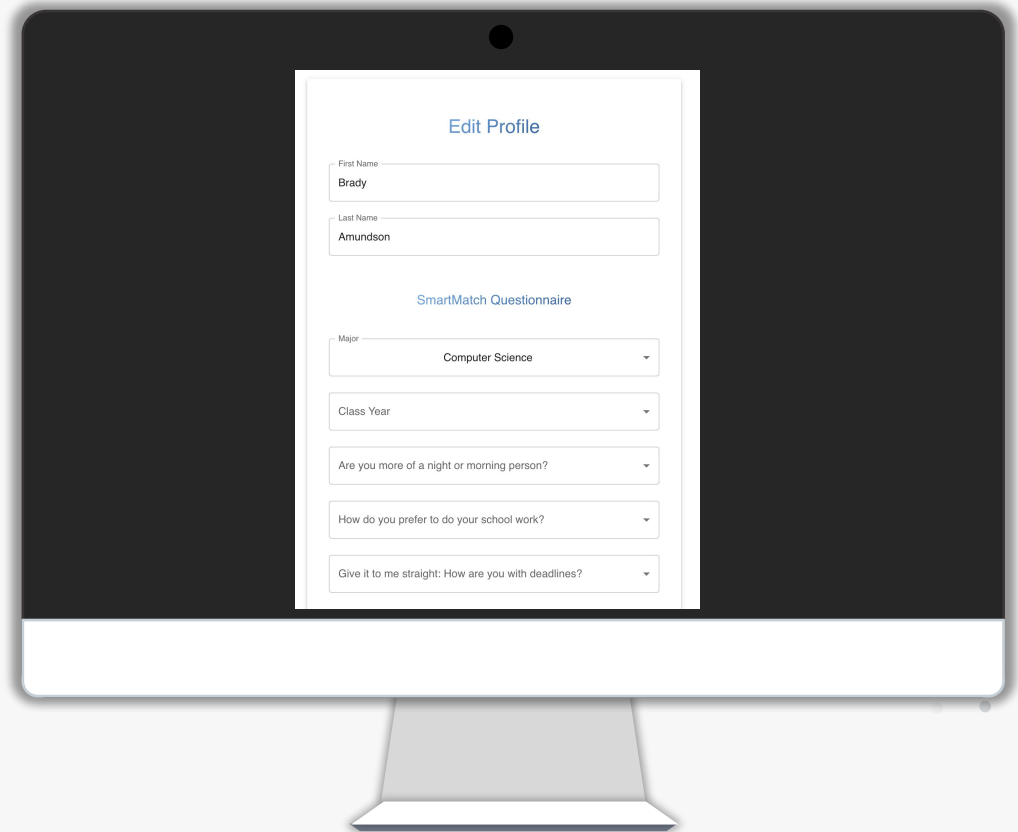
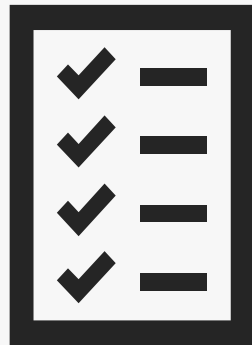
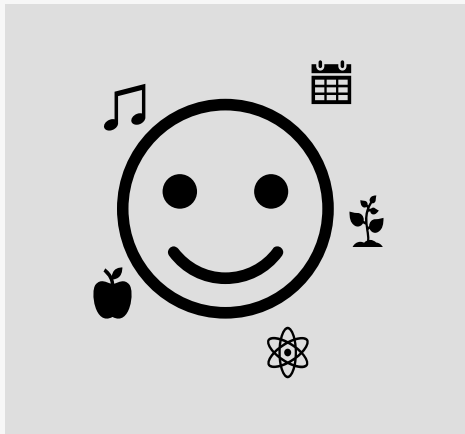
4

Intelligent  
SmartMatch  
Formation

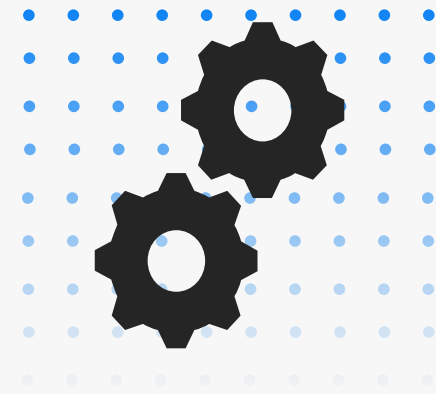
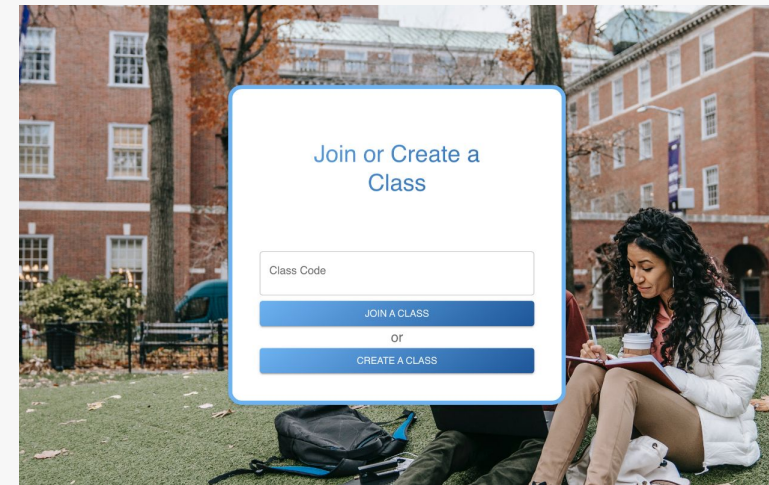


# Gruuper values personalization.

Students construct a profile specifically catered to them to ensure the most compatible groups are formed.



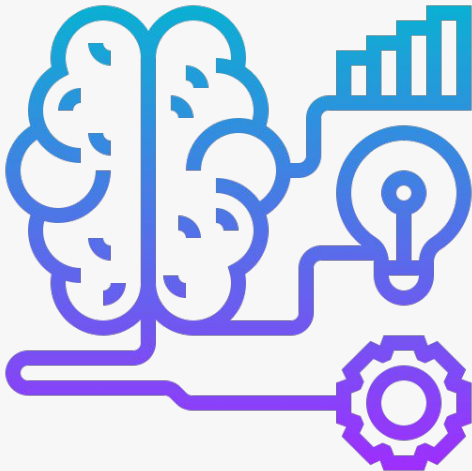
# With Gruuper, student matchmaking is a breeze.

A screenshot of the Gruuper student matchmaking interface. At the top left, there's a control for 'Maximum Group Size' set to '02', with '+' and '-' buttons. To the right are three icons: a crossed-out box, a group of people, and a document. Below these are four group cards labeled 'Group 1' through 'Group 4', each with a lock icon and a list of members. 'Group 1' has Bob Joe and John Wayne. 'Group 2' has CJ Plantemoli and Samuel Student. 'Group 3' has Peter Piper and Brady Amundson. 'Group 4' has George Washington. At the bottom is an 'Unmatched Members' box with the text 'Drop student here'.

Enter a class code to join a classroom. For Professors, simply click 'Create a Class'.

# Gruuper uses *Machine Learning* to make the best groups possible.

It's far too common for class groups to run into a variety of different problems. With ML, we can help with that.



```
return tf.nn.sigmoid_cross_entropy_with_logits(logits=logits, labels=labels)

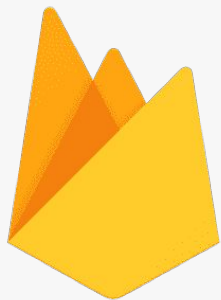
self.d_loss_real = tf.reduce_mean(
    sigmoid_cross_entropy_with_logits(self.d_logits, tf.ones_like(self.d_logits))
)
self.d_loss_fake = tf.reduce_mean(
    sigmoid_cross_entropy_with_logits(self.d_logits, tf.zeros_like(self.d_logits))
)
self.g_loss = tf.reduce_mean(
    sigmoid_cross_entropy_with_logits(self.g_logits, self.d_logits)
)

self.d_loss_real_sum = scalar_summary("d_loss_real", self.d_loss_real)
self.d_loss_fake_sum = scalar_summary("d_loss_fake", self.d_loss_fake)
self.d_loss = self.d_loss_real + self.d_loss_fake

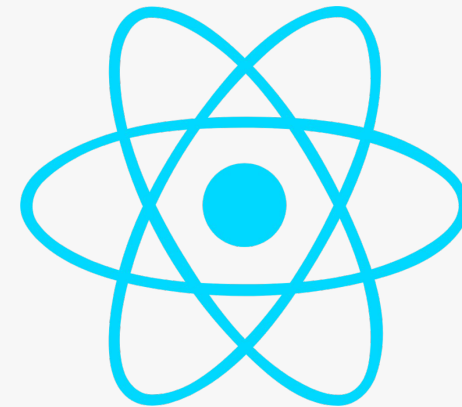
self.g_loss_sum = scalar_summary("g_loss", self.g_loss)
self.g_loss = self.g_loss
```



# Gruuper Specs



Firebase



# Live Demo!





# Thank you!

