

Using GitHub, Travis CI, and Python to Introduce Collaborative Software Development

Rowan Castellanos, Gregory M. Kapfhammer, Nicholas Tocci
PyCon 2018 Education Summit
May 10, 2018



Challenges

Existing tools support effective teaching

But, we needed a tool to solve extant concerns

Here are some of the key challenges that we faced!

Version Control is ... Only the First Step!

Version Control is ... Only the First Step!



Start a New
Repository



Populate the
Repository



Students
Copy Files

Version Control is ... Only the First Step!



Start a New
Repository



Populate the
Repository



Students
Copy Files



Grade the
Submission



Finalize a
Repository

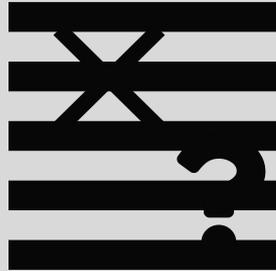


Students
Edit Files

Teaching Computer Science is Hard!



Unclear Status



Messy Code



Tricky Teamwork

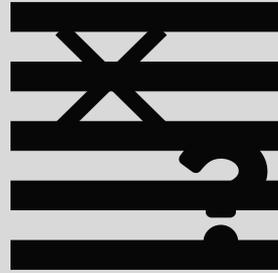


Exhausting Evaluation

Teaching Computer Science is Hard!



Unclear Status



Messy Code



Professor



Teaching Assistant



Tricky Teamwork



Exhausting Evaluation



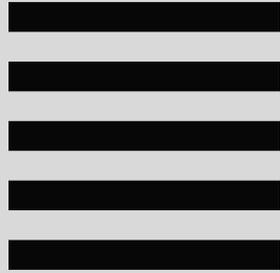
Student

What Are the Key Educational Goals?

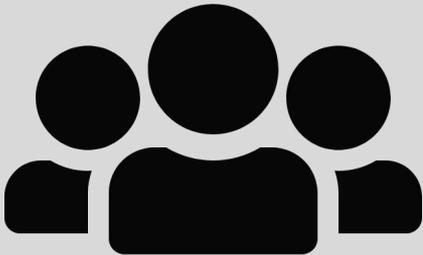
What Are the Key Educational Goals?



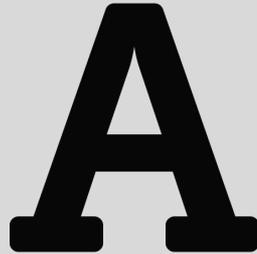
Clear Status



Standardized Code



Effective Teamwork

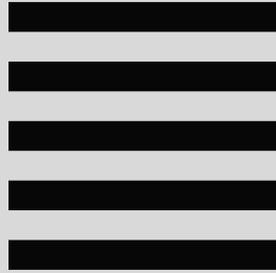


Efficient Evaluation

What Are the Key Educational Goals?



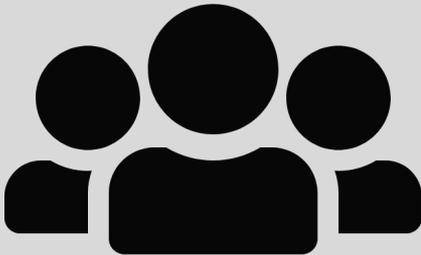
Clear Status



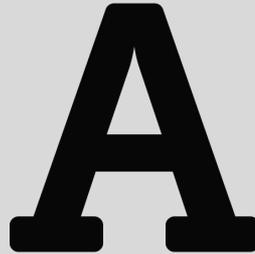
Standardized Code



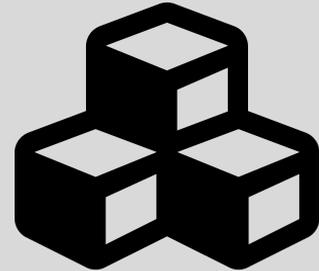
Critical Ideas



Effective Teamwork

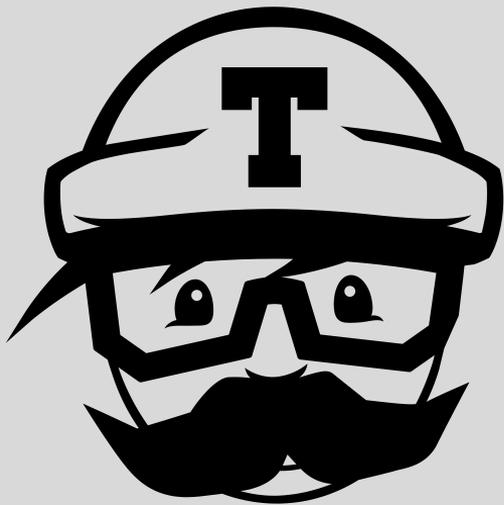


Efficient Evaluation



Industrial Tools

Tools Solve Some Issues ... Not Others!



Automate a Build + Check in the Cloud + Collaborate with GitHub

But, certain solution properties are difficult to check!



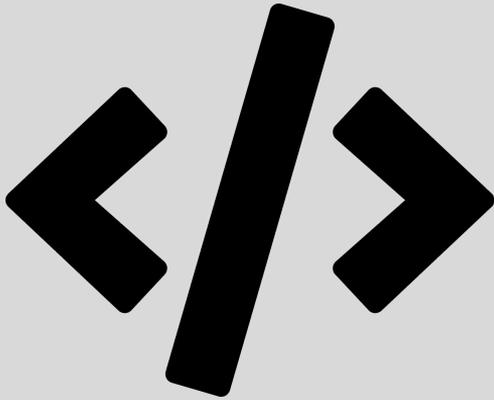
Solutions

**Adopt existing tools
whenever possible**

**Use Python language for
prototyping and testing**

**Here is how we created
the GatorGrader tool!**

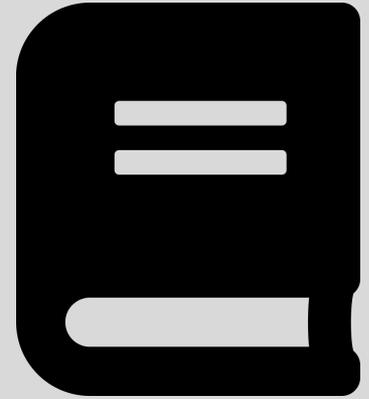
GatorGrader Checks Code and Writing



**Source
Code**



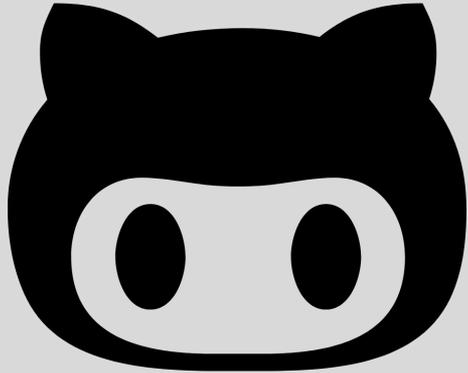
GatorGrader



**Technical
Writing**

Inspect the source code, program output, and technical writing

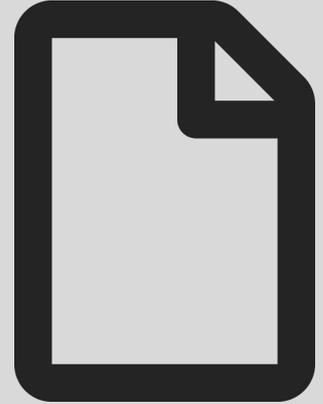
GatorGrader Can Check Other Entities!



**GitHub
Repository**



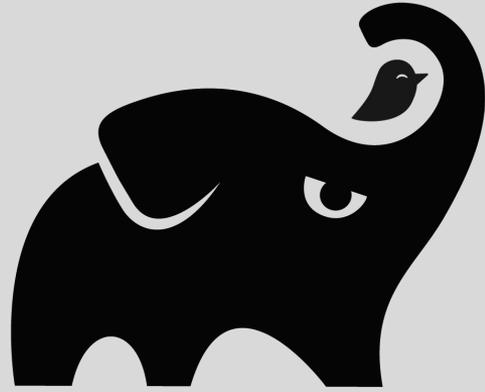
GatorGrader



**Data &
Images**

Ensure GitHub collaboration and the existence of arbitrary files

Performing Checks with GatorGrader



Gradle

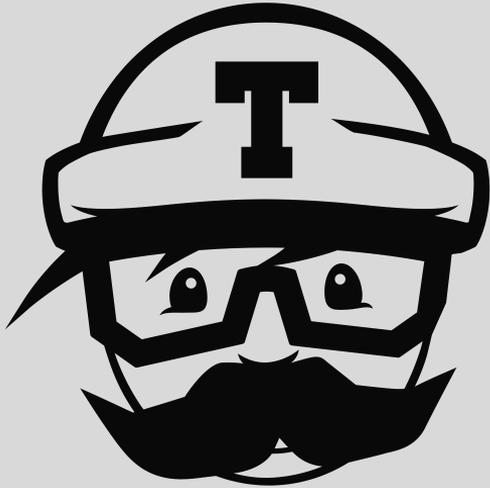


GatorGrader

Gradle plugin calls GatorGrader and gives feedback to a student

Supports dependencies between checks and parallel builds

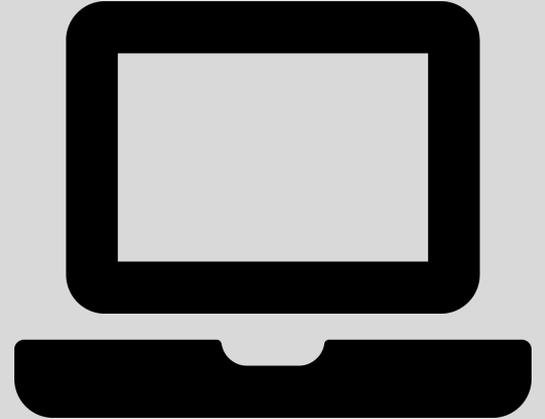
Where Does GatorGrader Run?



Travis CI



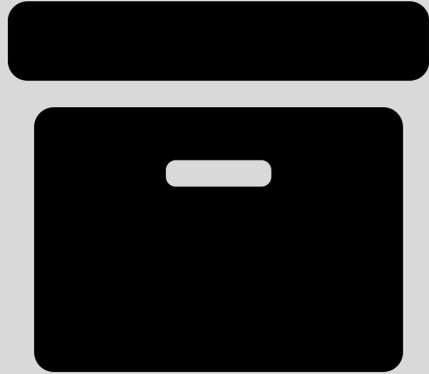
GatorGrader



Student Laptop

GatorGrader gives regular feedback both locally and in the cloud

Why Pick Python for Programming?



Packages



Versatility



Testing

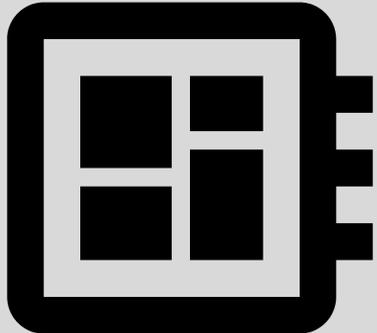
Many versatile packages support development and exploration

Using pytest leads to a confidence in tool correctness

What Courses Have Used GatorGrader?



**Computational
Expression**

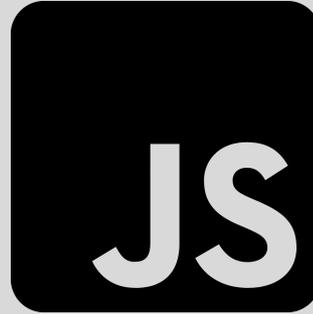


**Data
Abstraction**

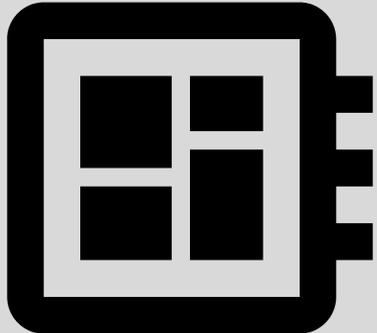
What Courses Have Used GatorGrader?



**Computational
Expression**



**Web
Development**



**Data
Abstraction**



**Software
Engineering**



Feedback

**GatorGrader promotes
technical skill mastery**

**Automated checkers can
become a limiting crutch**

**Increases confidence in
delivery of correct labs**



Janyl Jumadinova
Assistant Professor

Professor

GitHub Classroom, Travis CI, and GatorGrader made it easier for me to effectively deliver programming labs in an introductory computer science course!”



Saejin Mahlau-Heinert
Teaching Assistant

Assistant

This approach ensures that the source code and GitHub repositories are organized. It is easier for me to help the students who are struggling in an introductory course.”



Race Mahoney
Teaching Assistant

Assistant

You need to be really careful to ensure that students do not become overly reliant on the automated tools that check their source code and technical writing.”



Maria Kim
Teaching Assistant

Assistant

This tool suite made it easier for me to talk with students about technical requirements. It helped me to make complex assignments more accessible to students.”



Samantha Darris
Introductory Student

Student

GatorGrader encouraged me to add better code comments and try out language constructs that I would not have otherwise investigated. The tool was a big help!”



Anna Yeager
Introductory Student

Student

**GatorGrader is like
having a constant coach!
I liked receiving feedback
on the quality of my
source code and writing
before turning in the final
version of my lab.”**



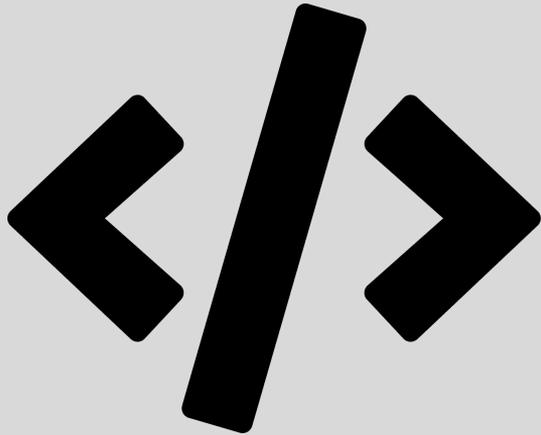
Conclusion

**GatorGrader aids the
building of positive habits**

**Python was the right
programming language**

**There are many avenues
for exciting future work!**

Experiments with GitHub & GatorGrader



Source
Code



GatorGrader

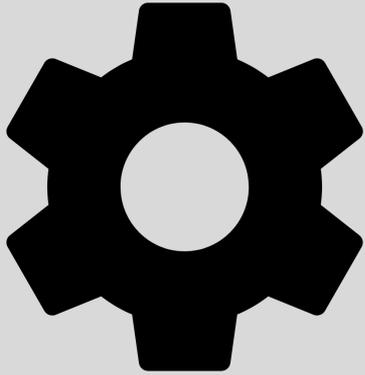


Novel
Insights

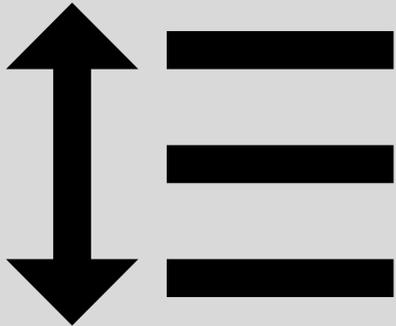
Experimentally study student writing, source code, and emotions

What source code and technical concepts lead to frustration?

Use and Contribute to GatorGrader!

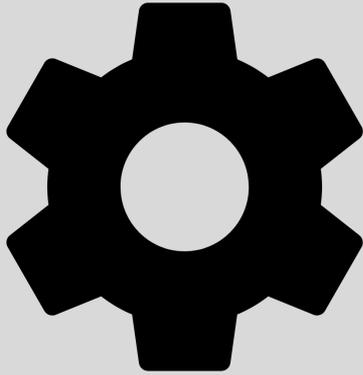


**New
Courses**

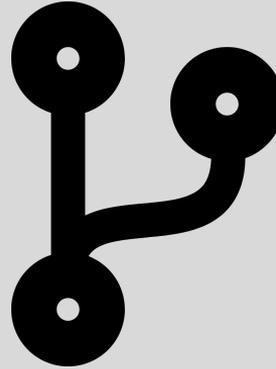


**Constructive
Feedback**

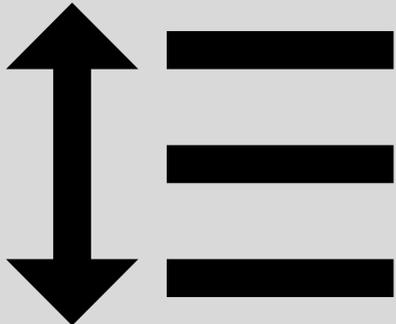
Use and Contribute to GatorGrader!



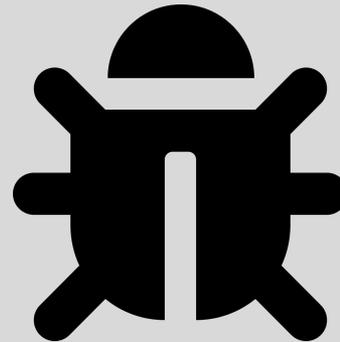
**New
Courses**



**Pull
Requests**



**Constructive
Feedback**

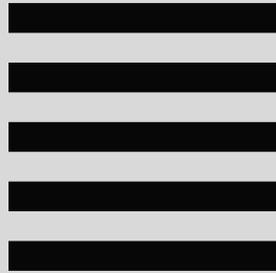


**Defect
Reports**

GitHub + Travis + Python + GatorGrader



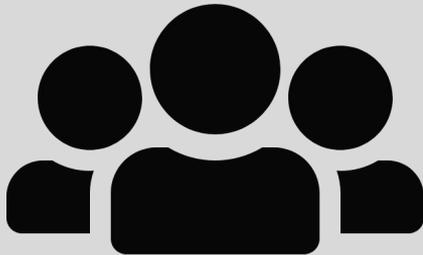
Clear Status



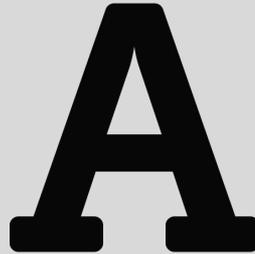
Standardized Code



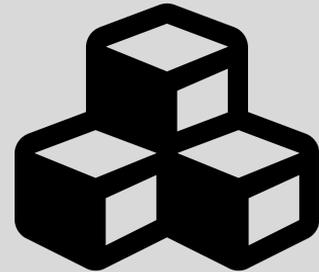
Critical Ideas



Effective Teamwork



Efficient Evaluation



Industrial Tools