CSG Auto Grader: Feasibility Report

Zachary Nelson, Ryan Hurst, Brandon Hora

Product

CSGrader will provide computer science educators and students with an advanced platform designed to streamline the educational process. This is accomplished through automated analysis and feedback provision for programming assignments, significantly reducing the time required for educators to grade assignments, and ensuring students receive timely, valuable feedback to enhance their learning experience.

Technical Feasibility

CSGrader is already capable of grading programs written in Java, C++, and .NET. The ongoing project is dedicated to expanding this service to include additional programming languages, such as R, Assembly, and MATLAB. Concurrently, the project seeks to augment the CSGrader platform with enterprise features that will improve the usability and maintenance of the platform. As much of the system is already built and working, proof of satisfactory technical systems is already present, and the project will simply be extending proven methods to new areas.

Social Feasibility

Of the several stakeholders impacted by the CSG Auto Grader (Students, Faculty, and Product Managers) the auto grader ensures fast and reliable feedback for students and faculty. Faculty will be able to use our tools to reduce time spent grading software projects, and students will enjoy reliable and accessible feedback. There will be training on the tools offered, especially with our points of contact within the Computer Science department. Most cooperation will be through our points of contact as faculty discover issues and changes to address.

Economic Feasibility

The efficiency and ease of mind the CSG Auto Grader brings is the primary benefit brought to faculty and their students. We expect production costs and maintenance costs would be outweighed by the value brought to these stakeholders. Benefits include increased production, and increased capacity for faculty. We expect these increased benefits will allow faculty and students better productivity working together. While the current investment/flow is not known, with an investment of $100000 and an average cash flow of $20000 per year, it would take $100000/$20000 = 5 years to break even.

Market Research

CSGrader is attractive to a niche, but continuously growing market. It may be useful to educators, bootcamps and even to firms onboarding new hires or providing ongoing training in supported languages. Other autograders are providing autograding for non-programming

assignments. Others offer in depth grading benefits. One of the most problematic parts of programming assignments remains testing and grading. Competition would rely on additional features that provide additional value to faculty and their students. Supporting additional languages such as R, Assembly or MATLAB provides this extra value.

Alternative Solution

Both alternative solutions of a full service class grader and a deeper, graduated grading experience are both rejected after careful consideration. Both solutions fall out of the time and scope of this project. Having a broad system for grading other types of homework and providing in-depth grading benefits is not realistically within our reach for this time frame. Our greatest chance for success is to narrow our scope and focus on making a simple but working autograder with more additions on later sprints.

Project Risks

Key risks for the CSG Auto Grader include false positives and negatives as a result of product use. False positives or negatives require review by faculty and reduce completion time. Other risks include new languages being adopted by faculty; if the language is not supported by the auto grader, completion time is increased. These risks will require input and cooperation with the Computer Science dept as well as continuous maintenance support.