

Grady – Grading AI System

Student: Celeste Amengual

Product Owner: Sam Ganzfried

Instructor: Dr. Masoud Sadjadi, Florida International University

PROBLEM

- There are two main grading approaches used by instructors. An absolute grading scale, which is fixed and divides grades into brackets. And a curve, which is assigned once numerical grades are finalized.
- The Grady system is designed to intelligently take a set of grades and provide a fair distribution to them.
- The Grady system is a preliminary prototype, there are many features that need still need improvement, optimization, and testing.

CURRENT SYSTEM

- Input: Number of grades, number of students, desired letter grade labels, grade intervals for each label & prior probabilities
- Output: each numerical score along with it's assigned letter grade
- Algorithm: Based on Bayesian Gaussian Mixture Model and Markov Chain Monte Carlo (MCMC) method. Integrates a Gibbs sampler with the expectation-maximization (EM) algorithm.
- Web application: Login page, dashboard (where inputs are submitted), results page (displays output)

REQUIREMENTS

- Ensure that runtime is efficient and reasonable.
- Ensure program produces accurate output on a variety of different grade distributions.
- Ensure that the software is scalable to large classes with 100 or more students.
- Made web-based version of the program more streamlined and easier to use for outside the project.

SYSTEM & OBJECT DESIGN

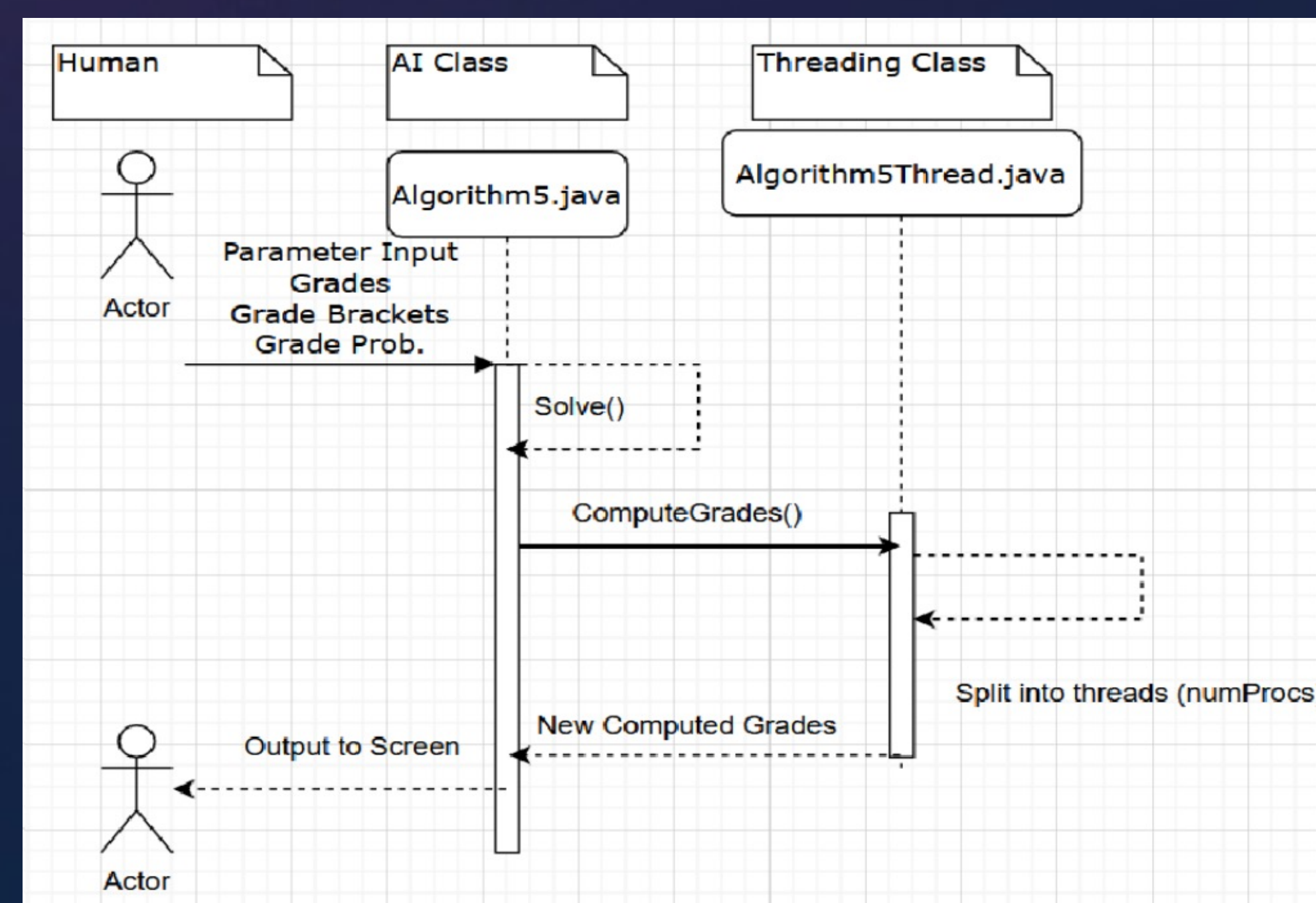


Figure 1 (above) Sequence Diagram representing flow of information within the software.

IMPLEMENTATION

- System uses PHP to link front-end and back-end
- JavaScript and bootstrap is used for the front end (adding dynamic UI and styling respectively)
- Java is used for implementing algorithm and optimizing assigned grades
- MAMP & XAMPP is used to set up a local server for testing purposes

SCREENSHOTS

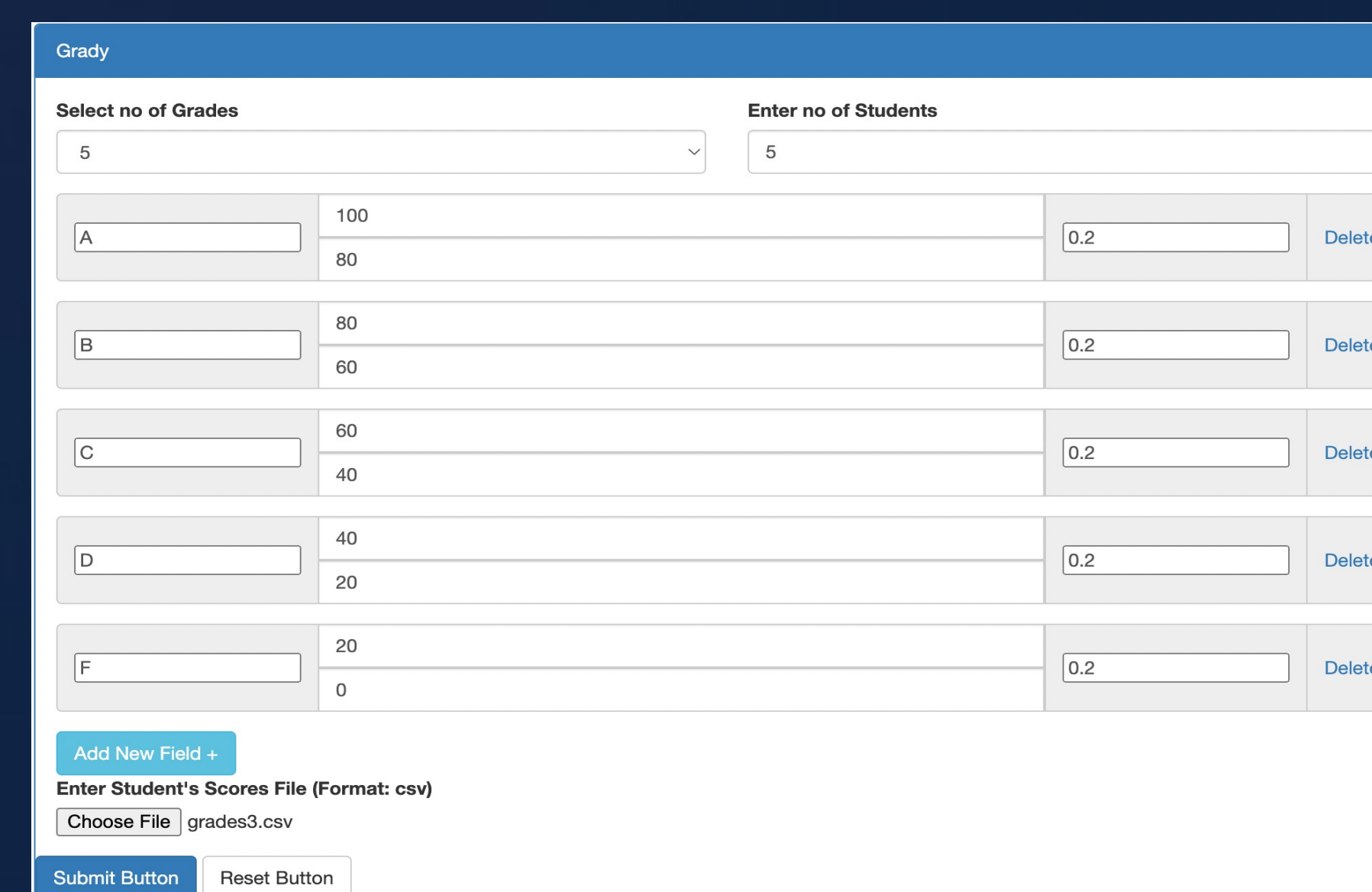


Figure 2 (above): Showcases the web application dashboard.

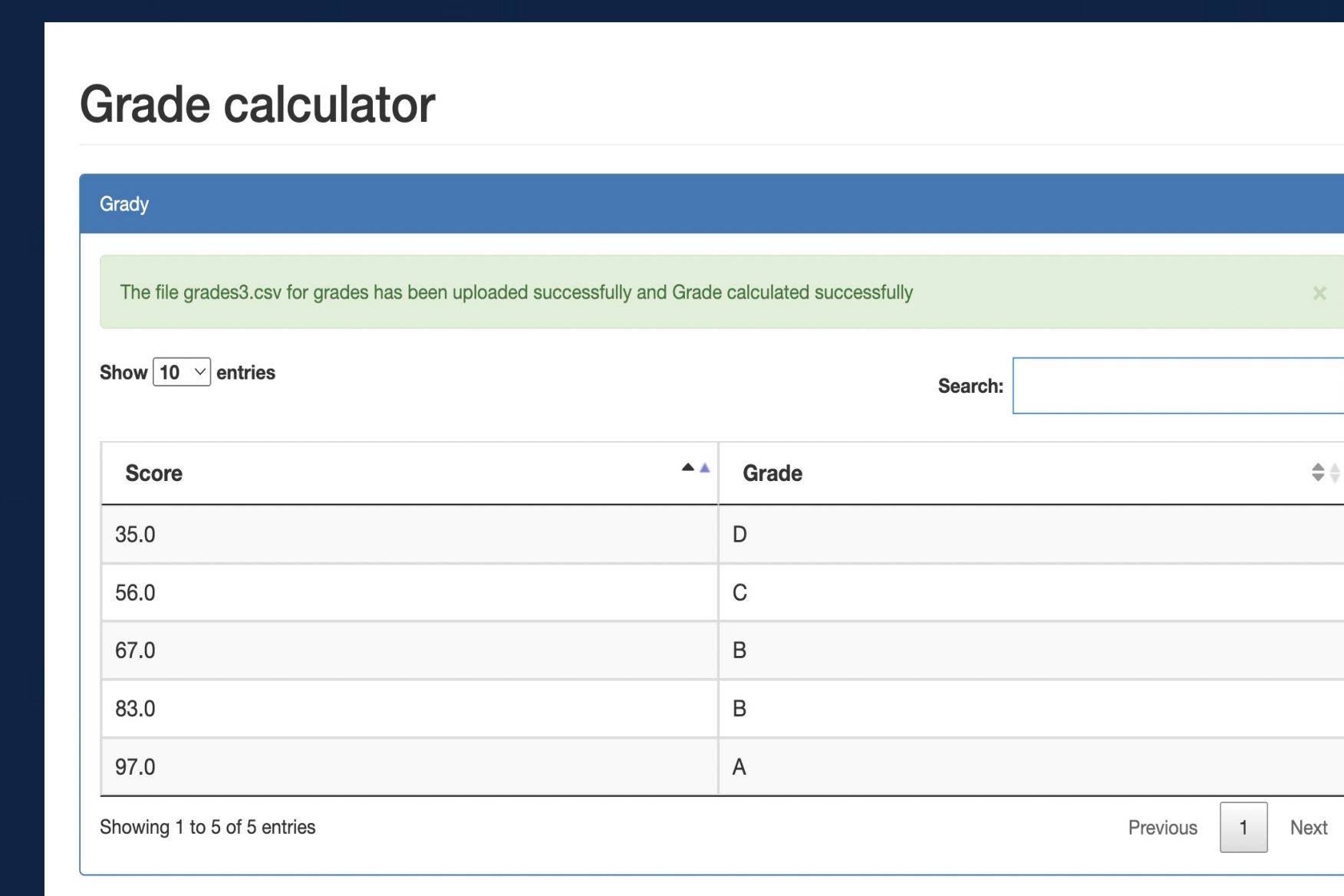


Figure 3 (above): Showcases the program running on the web application with an input of 5 students and 5 grades.

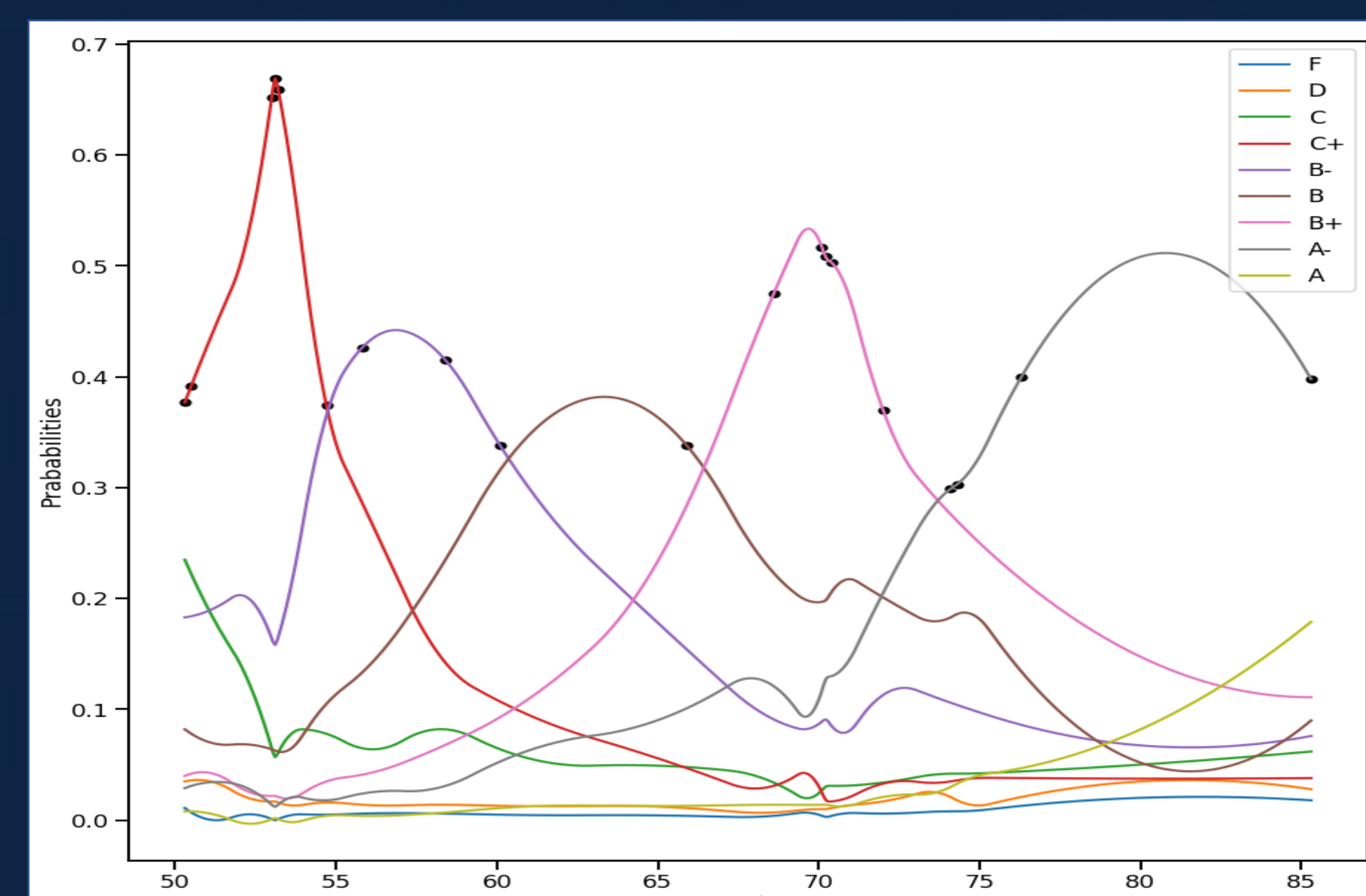


Figure 4 (above): Shows how the algorithm generates Gaussian curves which decide what letter to assign to each grade, all grades have a percentage for each letter and the letter with the highest percentage is the one we assigned to the grade

TECH USED



SUMMARY

- Grade Calculations were tested extensively. The runtime of the algorithm was consistent throughout, however optimization would be necessary for scalability.
- Improved User Interface allows for a better and more intuitive experience by providing more grading customization options
- Web app now uses PHP sessions and HTML local storage to allow users to edit previous submissions and to save time by avoiding repetition
- Experienced hurdles with machine competency to due to lack of documentation

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Sam Ganzfried. We thank Sam Ganzfried for their assistance and mentorship that we received throughout the senior design project.