Gaurav Srivastava • Naveen Ganessin • **Alex Rosenberg** • Michael Schloss

ABC

Project Backlog

Team Members

Gaurav Srivastava — srivast6@purdue.edu
Naveen Ganessin — nganessi@purdue.edu
Alex Rosenberg — rosenbea@purdue.edu
Michael Schloss — mschlos@purdue.edu

Problem Statement

With ABC, we hope to simplify classroom management and make an extremely responsive and aesthetically pleasing application. It will be super easy for a student, teacher, or administrator to use on any of their devices. By excluding auxiliary details such as discussion boards, ABC will keep the primary focus on classroom management rather than communication, which can be done through pre-existing systems.

Background Information

Classroom management is important to teachers, students, and administrators because systems for grading, submitting homework, and performing other classroom tasks should not get in the way of the education process.

There are similar popular learning management systems like Blackboard, Moodle, Edmodo and Desire2Learn. Since Purdue is most familiar with Blackboard, it is a good benchmark to compare ABC to. Unlike Blackboard, ABC uses native OS features such as Notification Center, Power Management, and Storage. By being built on top of Java, it can run on any OS at any time, making it super easy for a student, teacher, or administrator to use on any of their devices.

Another issue with apps like Blackboard is their tendency to be much too bloated with plenty of additional features which are not utilised much, with the features often implemented poorly. ABC aspires to move away from this and go back to the simple functionality of a learning management system by not loading the application with extra features and keeping the ones it has really easy and enjoyable to use.

Functional Requirements

ID	Requirement
1	As an admin, I can add/delete classes
2	As an admin, I can change class storage quota
3	As an admin, I can add/delete teachers
4	As an admin, I can add/delete students
5	As an admin, I can post school-wide announcements
6	As an admin, I can get course-wide or school-wide grade stats
7	As an Instructor, I can add/delete assignments, quizzes, and tests
8	As an Instructor, I can view file submissions from students
9	As an Instructor, I can set time-limits on assignments, quizzes, and tests
10	As an Instructor, I can post class-wide announcements
11	As an Instructor, I can send these announcements as emails should they wish
12	As an Instructor, I can host content on their class
13	As an Instructor, I can specify whether their hosted content is private or public
14	As an Instructor, I can teach multiple courses
15	As an Instructor, I can submit students' grades for assignments, quizzes, and tests
16	As an Instructor, I can view grades for any of their classes
17	As an Instructor, I can generate grade reports
18	As an Instructor, I should be able to generate a random text for students to enter on their account for attendance. (if time allows)
19	As a Student, I can see school-wide announcements
20	As a Student, I can see class-wide announcements
21	As a Student, I can see their grades from every class to which they're enrolled

ID	Requirement
22	As a Student, I can submit assignments
23	As a Student, I can take quizzes or tests
24	As a Student, I can send email to their teacher from the application
25	As a Student, I can view/consume public class files
26	As a Student, I should be able to receive an email notifications about assignments and exams due (if time allows)
27	As a Student, I should be able to enter the exact teacher's random text for attendance. (if time allows)

Nonfunctional Requirements

ID	Requirement
28	The application will be responsive
29	The application will use robust encryption to keep all data secure
30	Implement a caching feature that will allow users to see some data without an internet connection
31	The application will not use too much storage
32	The application will remain light on system resource usage
33	The application will communicate with native OS features to give the best possible experience
34	The application will have an aesthetically pleasing UI
35	The application will have separate UIs for different stakeholders for the app.
36	The database will have the proper keys and data types and enforce these restrictions
37	The database will have the correct foreign key relationships and will maintain these checks automatically and strictly
38	The database will be made of InnoDB tables to allow for fast indexing and recall of data
39	The database will use SSL to communicate with the application to provide secure endpoints for data transfer
40	The database will be password protected
41	The database will return JSON formatted data to the application
42	All passwords should be stored only as hashes