**CS 372 Group Project Proposal**

**Assignment Organizer**

**Fall 2020**

**Course Instructor –** Professor Tanzina Akter

**Group Members -** Shubham Mehta, Harpinder Minhas, Raju Chatelall and Shawn Fernandes

**Objective**

To build a cross-platform desktop utility app with cloud integration that allows students to backup and organize their assignments systematically in a hierarchical manner both locally and on the cloud.

**Introduction**

A vast majority of university students experience the chaos of organizing their assignments, exams, projects and reports for all their classes throughout their study. It is often hard to find old assignments from the previous courses if students have not organized their assignments systematically. Depending on different types of assignments and file types, a simple organizing task can end up being quite cumbersome. We identified this problem from our own experience and decided to build a solution for it. Hence the name of our app is “Assignment Organizer.”

**Functional Requirements**

Our goal is to design a cross-platform app with cloud integration that can be used by students to organize their assignments in a systematic order based on the semester-course hierarchy. The app will allow users to manage their assignments and store them locally on their hard drive and a cloud platform such as Google Drive, Dropbox, Microsoft OneDrive and others. Once a user has configured the app and uploaded their assignments based on the semesters and the courses, the app will organize everything and allow users to access this information from a single location. Such backup in a single location would save them time and hassle of organizing and searching for their assignments. In summary, our functional requirements are:

* The app should be able to run on Windows/macOS/Linux desktop environment.
* The app should allow a student to add a semester.
* The app should allow a student to add a course/class.
* The app should allow a student to upload and download single or multiple assignments for any given course.
* The app should provide its users with an option to back up their assignments on at least one of the cloud platforms such as Google Drive, Dropbox, Microsoft OneDrive and others using their account for the cloud platform service.

**Quality Requirements**

Our goal is not only to build an app that will provide an excellent experience to its users but also build an app that will be easy to maintain and extend. We will strive for the following quality requirements in the final product:

* The app should be functionally and logically correct. The app should implement all the functional requirements in a logically consistent manner.
* The app should save assignments successfully and organize it in an appropriate and hierarchically correct location, thus making it reliable.
* The app should be robust enough not to crash in any unanticipated circumstances. It should either keep running or quit gracefully after communicating the error to the user.
* The app should provide an easy to understand interface and require close to zero learning efforts by the student/user. The app should serve as a help to the student and not as a hindrance.
* The app will be built on top of the Electron platform and shall only use resources necessary to run effectively and remain performant.
* We will build the app using a web-based application framework. The app will be portable and capable of running on three different operating systems Windows, macOS and Linux, with minimal or no changes.
* The app will be interoperable as it will be able to save assignments locally and in the cloud. This way, it will integrate as well as co-exist with cloud services like Google Drive.
* We will use a disciplined approach with proper documentation to produce quality source code. The source code should be easy to understand, test, debug and maintain.

**List of Programming languages:**

* HTML
* CSS
* JavaScript
* JSON

**Software Tools & Environments:**

* Visual Studio Code
* Electron Framework with Node.js
* GitHub (git)
* Google Drive API