ACKNOWLEDGMENT

First and foremost, we would like to thank our Director Fr. Kinley D' Cruz, Principal Dr. Neena S. P. Panandikar and Head of Department Mr. Gaurang Patkar for providing all the help we needed.

The team put in a lot of effort on the project. However, it would not have been possible to complete it without the help of our faculties.

We are highly indebted to Miss Christina Barretto(Asst. Professor) for her guidance and constant supervision as well as for providing necessary information regarding the project and to our co-guides Mr. Vivek Jog and Miss Nisha Godhino for their guidance.

We would like to take this opportunity to thank our institution DON BOSCO COLLEGE OF ENGINEERING and our faculty members of the Computer Engineering Department who helped and supported us to go ahead with the project.

ABSTRACT

Colleges, schools and other educational environments have a lot of existing tools which help students and teachers interact and share resources. These tools, however, are developed to serve very specific and limited purposes and are very difficult to manage manually. Most of them fail to provide a single comprehensive platform for all necessary management operations. Moreover, these existing tools also fail to cater to the management and analysis of data and thus are not used for generating insights and management optimisation.

Our project aims to deliver a scalable web services framework which is easy to work with for both the developer and the customers. It will allow connecting students and teachers through subscription portals, by letting faculty moderators post content which can be accessed by the students and synced whenever an internet connection is available. Furthermore, we aim to build a real-time video conferencing platform using a lossy compression algorithm involving neural networks. Finally, we plan on integrating various existing services like Google Calendar and Drive, to ease automation efforts, reducing the need to manually manage multiple tools.

Chapter 1

Introduction

The essence of education is the ability for teachers to efficiently disseminate knowledge and obtain positive feedback from students, by using tools for a good teaching-learning experience. Educational environments have a lot of existing tools which help students and teachers interact and share resources. These tools, as will be explained further in this report, are developed to serve very specific and limited purposes, and are difficult to manage manually. They fail to provide a single comprehensive platform for all necessary operations. Moreover, these existing tools fail to cater to the management and analysis of data and thus are not used for generating insights and management optimisation. Furthermore, our worldwide internet infrastructure is lacking the reliability it needs to help people stay connected [2].

1.1 Problem Definition

Multiple e-learning platforms exist, each trying to solve a unique problem. Very few platforms function using an offline-first architecture, thus leading to really high dependency

on network infrastructure. It gets difficult for both students and teachers to manage their work, since they have to use multiple platforms, dividing their attention among each of them. Current e-learning platforms also provide very little in the way of automation, and lack a common, integrable interface.

Not all software platforms are built considering unreliable internet service. This dependency on a reliable internet by a majority of its users thus affects the entire network, and reduces quality of service for all internet users. We can solve this problem in many ways, like improving the network infrastructure, or building applications that responsibly use internet bandwidth. If unattended, issues of an unreliable system can include lesser productivity.

1.1.1 Existing Systems

There are a lot of e-learning systems and platforms available in the market but the most popular ones are Google Classroom and Moodle. The following shall be a brief overview of these popular e-learning platforms.

Google Classroom

To quite an extent google classroom managed to replace the traditional learning management system with no paper requirement, easy document sharing using google docs, easy assignment submission process and features to integrate with other Google services like YouTube, Drive and even Meet. We feel it still couldn't fully become the replacement for education system because of the following:

- **Difficult account management.** Suppose you want to upload an assignment onto the classroom for submission, which is currently on another google account, you will

need to logout of the current account, download the doc and sign in with the required Google account and then upload; quite a hassle.

- **Doesn't have an offline first design**, which eliminates the need of always being online to access the documents, as the documents can be downloaded in the user's local storage for a limited amount of time.
- **No live post updates**, i.e. you need to keep refreshing to view the recent updates.

One biggest pros of google education services is modularity. The G-Suite offers all these loosely coupled services like Meet, Classroom, Hangouts, and many others.

Moodle

Yet another popular e-learning platform and the best alternative to Google Classroom. Its pros include limited offline use for certain features. It has impressive features for uploading and downloading lecture notes, creating quizzes and tests, supports push notifications for both students and teachers, generating reports and many others. It has a backup, restore and import features, which turn out to be really useful for teachers. The teachers can also manage learners' profiles and setting enrollment keys, with role-based restrictions.

One of it's biggest cons however include poor scalability, due to its tightly coupled nature. This also causes issues with robustness, since a single module crash can cause the whole system to crash. No inbuilt video-conferencing functionalities, and it also becomes difficult to integrate a third party service due to the same tightly coupled nature.

It also offers a relatively poor mobile offering, with limited integration into third-party modules. There is a major learning curve for building and taking Moodle administration courses if you've never done it before.

Active Document Platform

The Active Document Platform(AD) [1] is a very useful platform that solves the problem of unreliable internet connectivity, by serving as an offline-first and distributed way of sharing course documents. Not being dependent on a centralised server on the internet allows it to operate offline as well, and it synchronizes with its main node whenever internet connectivity gets established. The best part about AD is its document organization, and distributed collaborative editing. Furthermore, AD uses SCORM (Sharable Content Object Reference Model), which is a collection of standards and APIs which help streamline connecting between other e-learning platforms which also use SCORM. One of its biggest cons however is limited functionality, and not very user-friendly.

1.1.2 Proposed System

Our project aims to deliver a scalable web services framework which is easy to work with for both the developer and the customers. It will allow connecting students and teachers through subscription portals, by letting faculty moderators post content which can be accessed by the students and synced whenever an internet connection is available. Furthermore, we aim to build a real-time video conferencing platform using a lossy compression algorithm involving neural networks. Finally, we plan on integrating various existing services like Google Calendar and Drive, to ease automation efforts, reducing the need to manually manage multiple tools.

1.2 Purpose of the Project

We plan on integrating various existing services so that the students need to use a single application interface, and are kept updated. We will work towards reducing network bandwidth usage, adopting an offline-first architecture wherever applicable. We will also add many optimizations, especially in video conferencing by intelligently encoding only the required features and intelligently transmitting it. We will also build interfaces to anonymously collect and analyze data which can be used to measure, manage various processes by finding out patterns that can help teachers improve their course material, and allowing us to focus on what matters, letting the computer automate various basic operations for us. We will also work to make our project scalable, robust, and flexible, allowing integrations to various third-party modules using our API.

1.3 Scope of the Project

Must be implemented

- Scalable and loosely coupled Web Services Framework.
- Connecting students and teachers through subscription portals (courses), which can be made private by virtue of an authentication code if needed.
- Allowing faculty and students to have specific roles, which are moderated by limited people.

Should be implemented

- Integrating various existing services, like Google Calendar, Drive, classroom, etc. by building web-hooks, which allow for easy automation. (by virtue of bots)
- A real-time video conferencing platform using an experimental lossy compression algorithm involving neural networks.

Could be implemented

- Having a marketplace for modules, where it is as easy as integrating by pressing install.
- Creating a simple student planner application that can help the student manage submissions, projects, and time in general (integrated with Calendar for ease of use)
- A system that can perform collection, monitoring, reporting, and long-term benefit analysis of student and employee attendance at the college. It can help students set their priorities.
- An online test tool that allows faculty to create tests and students can answer tests.

1.4 Report Organization

The current introductory section provides a brief introduction to each chapter.

Chapter 1: Introduction

This section focuses on the purpose and scope of the proposed system of FOCUSA.

Chapter 2: Literature Survey

This section describes the concepts and technologies used to develop the project.

Chapter 3: Software Requirement Specification

This section provides information about the specific requirement of the proposed system.

Chapter 4: System Design

This section describes the software lifecycle model, which will be used in developing the software. It also includes system design and detailed design.

Chapter 5: Implementation

This section deals with the implementation of the project where in the snapshots of each execution steps are shown.

Chapter 6: Conclusion

This section deals with the conclusion that can be derived after implementing the final System.

Bibliography

[1] Active Document, http://sensei.lsi.uned.es/ActiveDocument/.

[2] Carlos Iglesias. As internet access proves critical, we are missing targets to connect everyone. https://webfoundation.org/2020/04/covid-missed-targets/, April 2020.