A Project Report on

# Developing Data Analytics Support for Creative Learning Web Framework

Submitted in partial fulfillment of the requirements for the award
of the degree of

**Bachelor of Engineering**

in

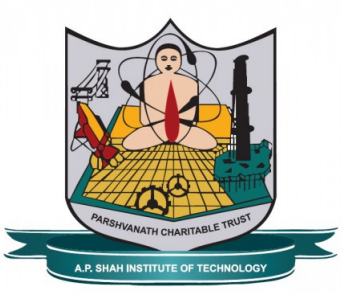
**Information Technology**

by

**Anand Morye (19104018)**
**Neel Dudheliya (19104049)**
**Ayush Jain (18104005)**

Under the Guidance of

**Prof. Jayshree Jha**




**Department of Information Technology**
**NBA Accredited**
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

**Academic Year 2022-2023**

# Approval Sheet

This Project Report entitled ***"Developing Data Analytics Support for Creative Learning Web Framework"*** Submitted by ***"Anand Morye"(19104018),"Neel Dudheliya"(19104049),"Ayush Jain"(18104005)***is approved for the partial fulfillment of the requirement for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

Prof. Jayshree Jha
Guide

Dr. Kiran Deshpande
Head Department of Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

# CERTIFICATE

This is to certify that the project entitled ***"Developing Data Analytics Support for Creative Learning Web Framework"*** submitted by ***"Anand Morye" (19104018) ,"Neel Dudheliya"(19104049),"Ayush Jain"(18104005)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***,to the University of Mumbai,is a bonafide work carried out during academic year 2022-2023.

Prof. Jayshree Jha
Guide

Dr. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

# Acknowledgement

We have great pleasure in presenting the report on **Developing Data Analytics Support for Creative Learning Web Framework.** We take this opportunity to express our sincere thanks towards our guide **Prof. Jayshree Jha** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards her constant encouragement, support and guidance through the development of project.

We thank **Dr. Kiran B. Deshpande** Head of Department,IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Sonal Jain** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

**Anand Morye**
**(19104018)**

**Neel Dudheliya**
**(19104049)**

**Ayush Jain**
**(18104005)**

## Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Anand Morye(19104018)

Neel Dudheliya(19104049)

Ayush Jain(18104005)

Date:

**Abstract**

Online learning has grown steadily in the last decades and the use of learning analytics has increased in parallel. As online education continues to grow, instructors need to find new ways to enhance student learning online and to understand students' interactions with their electronic learning environment. Learning Analytics or Web Analytics (WA) is predominately used to obtain key information about users and their behaviour on websites. Online learning consists of a Learning Management system (LMS) and Massive Open Online Course (MOOC). Online learning platforms have millions of users and a limited number of instructors. This results in no connectivity between the teacher and its students. Also, student behaviour on the platform remains unknown, and indeed, little research has been conducted into whether resources within the course are actually used properly. Knowing what the characteristics and behaviour of students are within the platform allows for changes to be made to existing courses, thus increasing their acceptance, improving the learning experience and helping to achieve the learning outcomes. Online learning platforms generate abundant amounts of data which can be utilised to monitor students performance on the portal. The aim of this research is to help instructors control the course content based on the student behaviour. In this project, we are going to implement data grabbing and visualisation in a MOOC system. Accordingly, a web analytic tool is selected based on its characteristics, functionalities and types. Web analytics tools support the business analyst's efforts in obtaining useful and relevant insights into market dynamics. Thus, generally speaking, selecting a web analytics and web metrics tool should be based on an investigative approach, not a random decision. Therefore, the use of Google Analytics (GA) is proposed to monitor student activity in MOOC, allowing for modelling the behaviour of students. It is also beginner friendly and easy to integrate. Once the student behaviour is analysed via GA, changes can be made to components and resources of MOOC and, thus increase the assent of students.

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| MOOC: | Massive Open Online Course |
| API: | Application Programming Interface |
| DOM: | Document Object Model |
| CICD: | Continuous Integration and Continuous Deployment |
| EDA: | Exploratry Data Analysis |
| LMS: | Learning Management System |
| LA: | Learning Analytics |
| UML: | Unified Modelling Language |

# Chapter 1

# Introduction

Online learning platforms have rapidly grown in popularity as a convenient and cost-effective alternative to traditional education. With the increasing demand for online education, instructors face the challenge of providing students with an engaging and effective learning experience in an online environment. To achieve this, instructors need to have a deep understanding of their students' behaviour and engagement while interacting with course content.

To address this challenge, this paper presents the design and development of an analytical tool aimed at providing instructors with insights into student behaviour and engagement in online learning platforms. The tool was built to collect data on students' interactions with course content, feedback, and performance, and to present this data in a visual format that is easy for instructors to understand and interpret. This tool provides instructors with a comprehensive analysis of student behaviour and engagement, allowing them to make informed decisions about their course content and structure, leading to improved student engagement and outcomes.

The goal of this research was to provide instructors with a practical and effective tool that could help them to better understand their students and improve the quality of education delivered through online learning platforms. The tool has the potential to significantly impact the quality of education delivered through online learning platforms, by providing instructors with valuable insights into student behaviour and engagement. The development of this tool represents an important step towards providing instructors with the tools they need to succeed in the rapidly growing online learning industry.

Furthermore, this analytical tool can provide instructors with a wealth of information about student behaviour, including: the average time spent on each section of the course, the number of students who complete the course, the most popular sections of the course, the average score on assessments, and the types of feedback provided by students. With this information, instructors can make informed decisions about the course structure and content, including which sections to keep or remove, which sections to modify, and which sections to add. By using this tool, instructors can improve student engagement and satisfaction, and provide a better overall learning experience for students.

## 1.1 Objectives

We intend to achieve the following objectives in our project.

- To gather and analyse data on student behaviour and engagement in online learning platforms to identify patterns and trends.

- To provide instructors with real-time insights into student behaviour and engagement, allowing them to make adjustments to their course content and structure in real-time.

- To present the collected data in a visual format that is easy for instructors to understand and interpret.

- To enhance the transparency and accountability of online learning platforms by providing instructors with datadriven feedback on their course content and structure.

- To support the development of personalised and adaptive learning experiences in online education, by providing instructors with data-driven insights into student behaviour and learning needs.

# Chapter 2

# Literature Review

The purpose of literature review is to gain an understanding of the existing research on Learning analysis on MOOC based platforms. This research helped us understand the current limitations on social interactions in online learning and how they affect student motivation and performance. It also helped us determine the data requirements.

- According to Prince, Michael Felder, Richard Brent, Rebecca [1], student engagement is a critical factor in the success of online courses and that it is essential to provide an engaging and interactive environment for students. The authors highlight the importance of designing course content that is engaging and relevant to students, using a variety of media, including videos, animations, and simulations. The paper provides specific recommendations for promoting student engagement, including the use of interactive activities such as virtual labs, case studies, and problem-based learning. The authors suggest that instructors should also provide opportunities for students to apply their learning in real-world situations and to connect with professionals in their field.

- Heilporn, Géraldine Lakhal, Sawsen Belisle, Marilou [2] examines the effects of different instructional strategies on student engagement in blended online courses. Blended online courses refer to a combination of online and face-to-face instruction, where students are engaged in a mix of online activities and traditional classroom activities. The purpose of the study was to determine which instructional strategies are most effective in promoting student engagement in blended online courses. The authors reviewed existing literature and surveyed students and instructors to gather data on the use of instructional strategies and the level of student engagement. The results showed that instructional strategies that promote interaction and collaboration, such as discussion forums and group projects, are most effective in promoting student engagement. Additionally, the use of technology, such as video lectures and multimedia presentations, was also found to enhance student engagement. The authors concluded that instructional strategies that promote interaction, collaboration, and technology use are crucial in promoting student engagement in blended online courses.

- Leitner, Philipp Maier, Karin Ebner, Martin [3] discusses the use of web analytics as an extension for a learning analytics dashboard on a massive open online platform. The authors highlight the need for a comprehensive learning analytics dashboard that provides instructors with real-time insights into student behaviour and performance. The authors propose that web analytics can be used to enhance the functionality

of a learning analytics dashboard, providing additional data on student engagement, interaction, and behaviour. The authors present a case study of a massive open online platform that integrated web analytics with a learning analytics dashboard to provide a more comprehensive view of student activity. The results showed that the use of web analytics in conjunction with a learning analytics dashboard improved instructors' understanding of student behaviour and allowed them to make more informed decisions about course design and delivery. The authors conclude that web analytics can be an effective tool for improving student engagement and outcomes in massive open online platforms.

- Dhumantarao, Thammi Raju Murthy, G.R.K. Khade, Shrikant Padmaja, B. Bs, Yashavanth Kumar, S. Soam, Sudhir Ch, Srinivasrao [4] explores the use of learning analytics to understand learner behaviour in online courses. The authors note that online courses offer a wealth of data on student behaviour, but that this data is often underutilized. The authors argue that learning analytics can be used to make sense of this data and provide insights into how students engage with course content, interact with their peers, and use technology. The authors present a case study of an online course that used learning analytics to track student behaviour and provide feedback to instructors on areas where students struggled or engaged. The results showed that learning analytics provided valuable insights into student behaviour, allowing instructors to make data-driven decisions about course design and delivery. The authors conclude that learning analytics can be a valuable tool for improving student engagement and outcomes in online courses.

- Gray, Julie Diloreto, Melanie [5] focuses on analysing the cognitive engagement of students in e-learning discussion forums through content analysis. The authors argue that cognitive engagement is an important factor in e-learning, as it is related to students' motivation, learning outcomes, and satisfaction. To examine students' cognitive engagement, the authors conducted a content analysis of students' participation in discussion forums in an online course. The authors identified several dimensions of cognitive engagement (e.g. critical thinking, creativity, collaboration) and used these dimensions to analyse the content of students' posts in the discussion forums. The authors found that students demonstrated varying levels of cognitive engagement in the discussion forums and that this engagement was related to their learning outcomes. The authors conclude that content analysis can be a useful tool for examining students' cognitive engagement in e-learning environments and suggest that further research is needed to examine the relationship between cognitive engagement and learning outcomes in online courses.

- İ. Yıldırım and S. Çırak-kurt [6] conducts a systematic review of existing research to investigate the impact of gamification on student engagement in online learning. The authors analyse studies that use gamification techniques, such as rewards and challenges, in online courses and evaluate their effectiveness in increasing student engagement. The results of the review suggest that gamification has a positive impact on student engagement in online learning, particularly in terms of increasing motivation and interest. However, the authors note that more research is needed to fully understand the impact of gamification on student engagement and to identify the most effective gamification strategies for different types of online courses.

- X. Solé-Beteta, J. Navarro, B. Gajšek, A. Guadagni, and A. Zaballos [7] focuses on using data-driven methods to quantify and measure the level of engagement of students in synchronous virtual learning environments. The authors aim to develop a robust model that can accurately capture student engagement and use this information to improve the quality of online learning experiences. The study uses a combination of log data and self-reported surveys to gather information on students' engagement levels during virtual learning sessions. The authors then analyse the collected data and develop a model that incorporates various engagement indicators such as participation, interaction, and motivation. The results show that the model is capable of accurately measuring student engagement in synchronous virtual learning environments and can be used to inform the design and delivery of online courses.

- Altuwairqi, K., Jarraya, S.K., Allinjawi [8] discusses the importance of measuring engagement levels of students in online learning environments. The paper proposes a framework that incorporates learning analytics to analyse student behaviour and engagement in online courses. The framework uses a variety of data sources, including clickstream data, forum posts, and grades, to measure engagement levels. The paper highlights the benefits of using learning analytics for student engagement, such as identifying at-risk students and providing targeted interventions, and also discusses the challenges of implementing learning analytics in higher education, such as data privacy concerns and ethical considerations. Overall, the paper provides a valuable contribution to the field of online education and learning analytics by highlighting the importance of student engagement and providing a framework for measuring it.

- Czerkawski, B.C., Lyman, E.W [9], discusses an instructional design framework to enhance student engagement in online learning environments. The authors highlight the challenges that educators face while designing and delivering effective online courses and the importance of engaging students for their success in such environments. The paper discusses the key components of the proposed framework, including the course content, instructional strategies, and technology used to create the course. The authors also discuss the importance of understanding the learners' characteristics and motivation for learning, and how this can be leveraged to design more effective online courses.

- R. K. Yadav, A. Singh, and M. Arora discusses the latest data analytics techniques for business intelligence, including machine learning, deep learning, natural language processing, and big data analytics. The paper highlights the challenges associated with data analytics such as data quality, data privacy, and data security. It proposes a framework for integrating data analytics into business intelligence systems, emphasizing the key considerations for successful implementation. The paper provides valuable insights for researchers, practitioners, and decision-makers interested in data analytics and its applications in business intelligence. The authors provide a comprehensive review of the latest data analytics techniques for business intelligence, highlighting their benefits and drawbacks. The paper also discusses how these techniques can be used to extract insights from the ever-increasing volume and complexity of data being generated by businesses. It emphasizes the importance of data quality, data privacy, and data security in data analytics. The authors provide a detailed framework for integrating data analytics into business intelligence systems. The paper also highlights the

key considerations for successful implementation of data analytics. Overall, the paper provides a valuable resource for those interested in data analytics and its applications in business intelligence.

- Tuti Purwoningsih,Harry B. Santoso,Zainal A. Hasibuan describes a study that uses exploratory data analysis (EDA) techniques and machine learning algorithms to develop models. The study combines student demographic profile data with student activity data in a learning management system (LMS) to analyze their relationship with student performance. The data cleaning and exploration phase is described as a stage that requires more resources and time to complete due to various data sources and formats. The study then goes on to describe the descriptive analytics of student activity in e-learning, student profiles, and distribution of student learning outcomes. The results are presented in tables and graphs, which provide insight into the diversity of student activity and profiles, as well as their learning outcomes.This article discusses the advantages and challenges of eLearning compared to traditional face-to-face learning. While eLearning offers greater flexibility and lower time requirements for students, its effectiveness varies across age groups and locations. To optimize eLearning outcomes, instructional design developers must adapt to the characteristics of eLearning participants, and data analytics can help in this process. Learning analytics (LA) is a promising approach to processing and analyzing large amounts of data collected from eLearning environments to improve learning outcomes. The study presented in this article uses exploratory data analytics and machine learning techniques to analyze the demographic and learning activity data of eLearning students. The results provide instructional design developers with recommendation materials to develop effective and efficient e-Learning instructional designs to support student success. Overall, the article highlights the importance of data analytics in developing student-centered eLearning strategies and improving eLearning outcomes.

# Chapter 3

# Project Design

## 3.1 Problem Definition

The problem that this project aims to address is the lack of visibility and insights into student requirements in online learning platforms. Currently, instructors are often unable to see how students are interacting with their courses, what aspects of the course are most popular, and where students may be struggling. This lack of information makes it difficult for instructors to improve their courses and better engage their students.

One of the main challenges in this area is the sheer amount of data that is generated by online learning platforms, and the difficulty in making sense of this data in a way that is useful for instructors. The data is often stored in disparate systems and is not easily accessible or comparable, making it difficult to draw meaningful insights. Furthermore, the traditional methods of collecting data on student interactions, such as surveys and questionnaires, can be time-consuming and often result in low response rates. This makes it difficult to get a comprehensive picture of student behaviour and limits the ability of instructors to make decisions about how to improve their courses. By building an analytical tool to capture and analyse data on student behaviour and engagement in online learning platforms, this project aims to provide instructors with the insights and recommendations they need to improve their courses and better engage their students. The tool will make it easier for instructors to see how students are interacting with their courses, what aspects of the course are most popular, and where students may be struggling. This will help instructors to make data-driven decisions about how to improve their courses and create a more engaging and effective learning experience for their students.

## 3.2 Proposed System

The initial step of data collection requires DOM manipulation and event management to capture student interactions on the website. Javascript is the predominant language used for the front-end task. The collected data is then parsed at the backend and stored in a database for further analysis and reporting. To be able to effectively store the data in the backend, frameworks such as Node.js and MongoDB are employed for the purpose of developing and deploying secure APIs. For our instructor dashboard, we are using chartjs, a JavaScript library that provides dynamic and interactive visualizations.

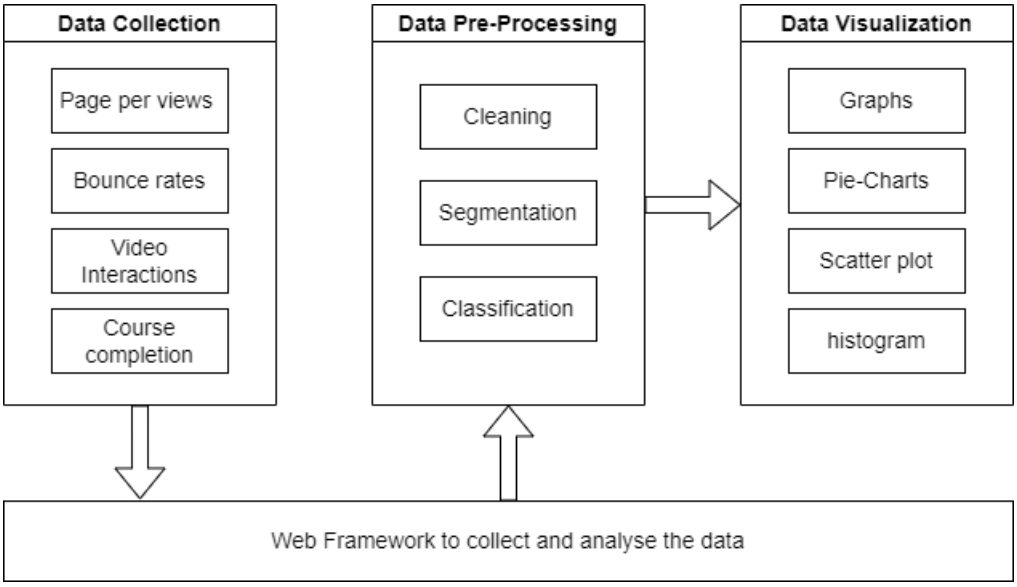



Figure 3.1: Proposed Architecture

- **Data Collection**: The first step is to collect data on student behaviour and engagement in online learning platforms. This data can be gathered through various methods, including tracking student activity within the platform, gathering feedback from students through surveys and questionnaires, and analysing student performance on assessments and other course-related activities.

- **Data Cleaning and Pre-processing**: Once the data is collected, it must be cleaned and pre-processed to ensure that it is ready for analysis. This may involve removing any irrelevant or redundant data, transforming the data into a format that is suitable for analysis, and normalizing the data to ensure that it is consistent and comparable across all students and courses.

- **Data Analysis**: The next step is to analyse the data to identify patterns and trends in student behaviour and engagement. This can be done through a variety of methods, including descriptive statistics, inferential statistics, and visualization techniques.

- **Data Visualization**: The results of the data analysis must then be visualized in a way that is easy for instructors to understand and interpret. This can be done through the use of various visualization techniques, such as bar graphs, pie charts, line graphs, and scatter plots.

- **Recommendations and Feedback**: Based on the results of the data analysis, the tool will provide instructors with recommendations and feedback on their course content and structure. These recommendations will be based on the data collected and analysed, and will help instructors to make informed decisions about how to improve their courses and better engage their students.

## 3.3 Metrics for Data Collection

Some of the metrics that we used to measure student interactions with the course materials are provided below. Additional metrics might be implemented based on requirements.

- Course Completion Rates: One of the most straightforward ways to measure student engagement is by tracking the number of students who complete a course. This can be done by tracking the percentage of students who complete the course compared to the total number of students who enrolled.

- Time Spent on Course Materials: Another way to measure student engagement is by tracking the amount of time students spend on different sections of the course. This information can be used to identify which parts of the course are the most engaging and which parts may need improvement.

- Student path through course modules: The path that the student takes to complete the course may differ from the traditional path set by the instructor. Because the student might already know the basics and want to learn advanced modules, he skips the initial topics. This can help instructors determine how to set the difficulty level of the course to better match the target audience.

- Video Playback Rates: In online learning platforms that incorporate video content, tracking video playback rates can provide valuable insights into student engagement. This information can be used to identify which parts of the video are the most engaging, and which parts may need improvement.

- Progress Tracking: By tracking the progress of students through the course, instructors can gain insights into which sections of the course are causing difficulties for students. This information can be used to improve the course design and content to make it more engaging and accessible to students.

- Analytics: Normal metrics, but for course pages, such as page views per visit, bounce rate, and user interactions with websites This will make it easier to give educators more individualized analytics.

## 3.4 System Diagram

### 3.4.1 Use Case Diagram

A use case diagram is a type of UML (Unified Modeling Language) diagram that represents the interactions between a system and its users or actors. It is a visual representation of the functional requirements of the system and the actions that users or actors perform when using the system.Use case diagrams consist of actors, use cases, and relationships between them. Actors are the people, systems, or other entities that interact with the system. Use cases represent the specific actions or tasks that actors perform when using the system. The relationships between actors and use cases are depicted through lines and arrows.Use case diagrams help to identify and clarify the requirements of the system and provide a high-level view of the system's functionality. They are useful for communicating with stakeholders and for guiding the development of the system. Use case diagrams can also help to identify potential errors or issues in the system design.In summary, use case diagrams provide a graphical representation of the interactions between a system and its users or actors. They help to identify the functional requirements of the system and provide a clear overview of the system's functionality. Use case diagrams are a valuable tool for software developers, project managers, and stakeholders in ensuring the successful development and implementation of a software system.

The use case diagram contains two primary actors: the Instructor and the Students.The students engage with the courses and feeds data to system about the usage of the course material.The use case diagram demonstrates the interaction between the actors and the system, highlighting the functions that the system can perform for each user. The Instructor can collect data on student behavior, access analytic to gain insights, and view student progress, while the Students can take courses, submit assignments, and view their progress.

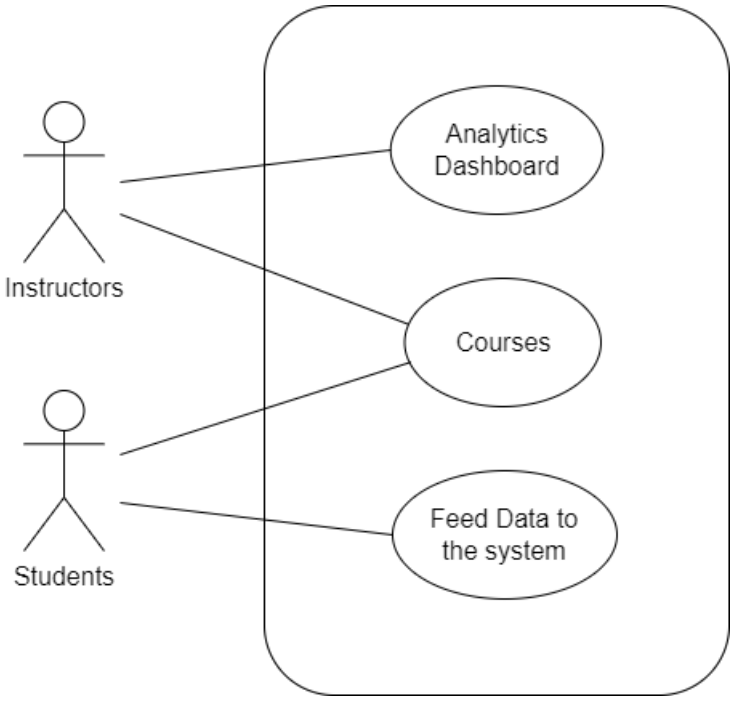



Figure 3.2: Use Case Diagram for Analytic Tool

### 3.4.2 Sequence Diagram

A sequence diagram is a type of UML diagram that shows the interactions between objects or components in a system over time. It is used to illustrate the dynamic behavior of the system, particularly how different components interact with each other to achieve a specific goal.In a sequence diagram, each object or component is represented as a vertical lifeline, and the interactions between them are shown as horizontal arrows. The sequence of events is shown chronologically from top to bottom, with time flowing downwards.Sequence diagrams are useful for visualizing complex interactions between different components of a system and identifying potential issues or bottlenecks in the system's design. They can also be used to test the behavior of a system under different scenarios or conditions.In summary, sequence diagrams provide a visual representation of the interactions between components of a system over time. They help to identify and analyze the dynamic behavior of the system, and can be used to test the system's behavior under different scenarios. Sequence diagrams are a valuable tool for software developers, system designers, and project managers in ensuring the successful development and implementation of a software system.

The sequence diagram for the web-based analytical module project depicts the interaction between the user interface, the JavaScript code that collects user data, the NodeJS (Express) backend, and the Mongo DB database. The diagram begins with the user accessing the MOOC platform and interacting with the analytical module to collect data. The module then sends the data to the backend using JavaScript, and the backend stores the data in the Mongo DB database. To retrieve the data, an API is created in ExpressJS which feeds the data to the instructor's dashboard, where visualizations are displayed. The sequence diagram shows the order in which these events occur, along with the messages or calls exchanged between the different components. Overall, the sequence diagram helps to illustrate how the various parts of the system work together to collect, store, and display user data.

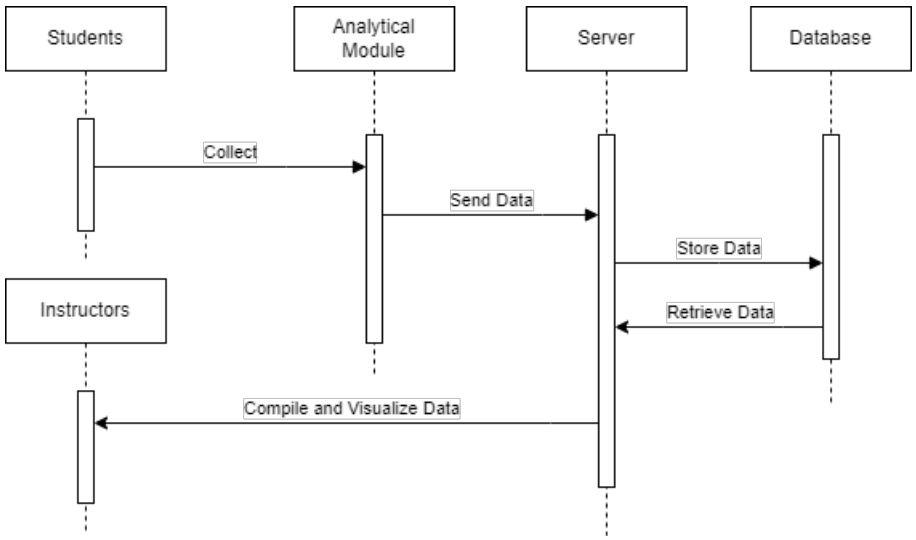



Figure 3.3: Sequence Diagram for Analytic Tool

# Chapter 4

# Project Implementation

### 4.0.1 Front-End

The course details page and its respective modules make up the front end of the project. The figures below depict the files responsible for collecting data from these pages. The below file is the Analytical module that collects data from the user and sends it to the backend.

```
import {Kratos} from "./kratos.js"


// Initialize unique user hash
const options = {
  dataFrequency: 5000,
  url: 'http://localhost:5000/analytics/record-userdata'
}
const kratos = new Kratos(options)
kratos.set("courseId", "641c684706ec5d6413405853");
// kratos.print()
// setTimeout(() => kratos.print(), 5000)
// setTimeout(() => kratos.sendData(), options.dataFrequency)
if(window.location.pathname == '/Achieving_Personal_and_Professional_Success.html'){
  const courseSidebar = document.querySelector('#course-left-sidebar')
  const courseContent = document.querySelector('#course-content')
  kratos.listenerAndFindOnce('click', courseSidebar, ['enrollButton'])
  kratos.listenerAndFindOnce('mouseover', courseSidebar, ['courseDetails'])
  kratos.listenerAndFindOnce('mouseover', courseContent, ['courseDescription', 'review'])
}


if(['/introduction.html' , '/module1.html' , '/module2.html' , '/module3.html' , '/module4.html']
.indexOf(window.location.pathname) + 1){
  const video = document.querySelector('#video')

  kratos.listenerforVideo('seeked', video )
  kratos.listenerforVideo('ratechange', video)
  kratos.listenerforVideo('pause', video)
  kratos.listenerforVideo('play', video)
  kratos.listenerforVideo('ended', video)
}
```

Figure 4.1: Data Analytic Module

Method for collecting metrics for video interactions in individual modules.

```
listenerforVideo(trigger, element, cb){
  this.data.video = {
    play: [],
    paused: [],
    seeking: false,
    seeked:[],
    rateChange: [],
    ended: false
  };
  element.addEventListener(trigger, (e) =>{
    switch (trigger) {
      case 'play':
        this.data.video.play.push(Math.round(e.target.currentTime));
        if(this.data.video.seeking){
          this.data.video.paused.pop();
          this.data.video.play.pop();
          this.data.video.seeking = false;
          this.data.video.seeked.push(Math.round(e.target.currentTime))
        }
        break;
      case 'pause':
        this.data.video.paused.push(Math.round(e.target.currentTime));
        break;
      case 'seeked':
        this.data.video.seeking = true;
        break;
      case 'ratechange':
        this.data.video.rateChange.push(e.target.playbackRate)
        break;
      case 'ended':
        this.data.video.ended = true;
        console.log('ended');
      break;
      default:
        cb();
        break;
    }
    this.updates = true;
  })
}
```

Figure 4.2: Method for Video Event Capturing

Modular Method that collects data based on the events and elements feed to it. It supports event delegation.

```
listenerAndFindOnce(trigger, element, arr) {
  if(!element && !trigger && !arr.length) {return false}
  const listener = (event) => {
    const id = event.target.id;
    arr.forEach((query) => {
      if(query == id){
        this.data[id] = true;
        console.log(this.data);
        this.updates = true;    // for update status
      }
    })
  }
  element.addEventListener(trigger, listener)
}
```

Figure 4.3: Method for Event Capturing

Course Details page.

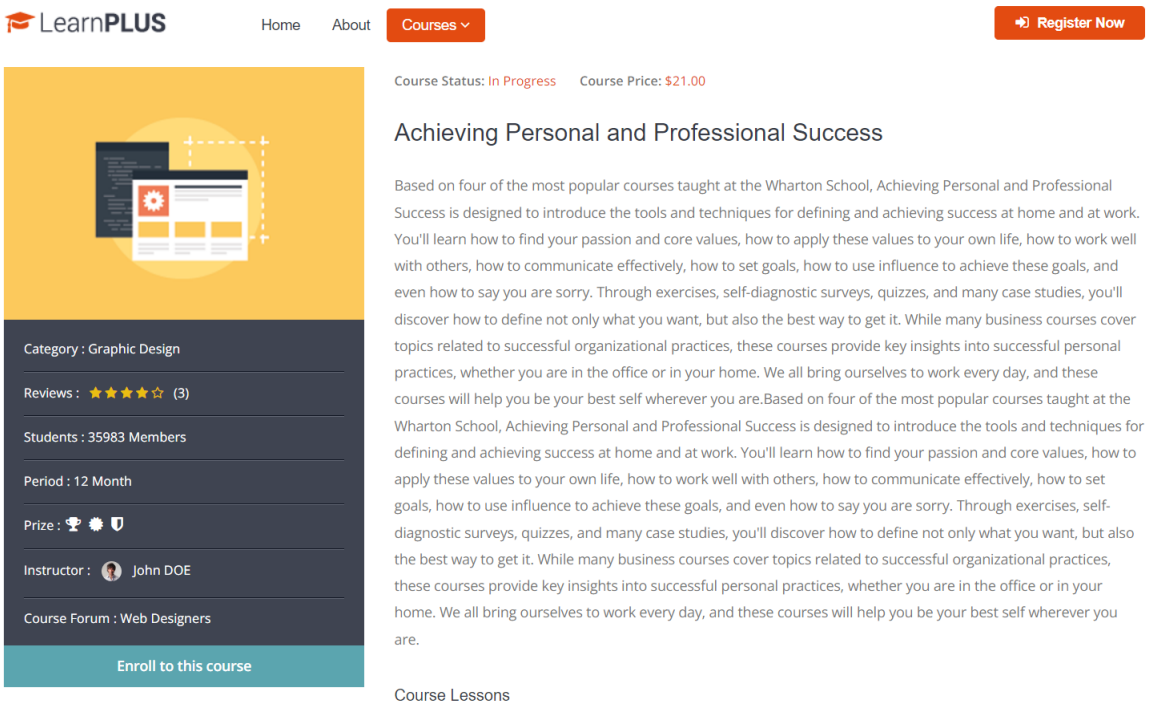



Figure 4.4: Course Landing Page

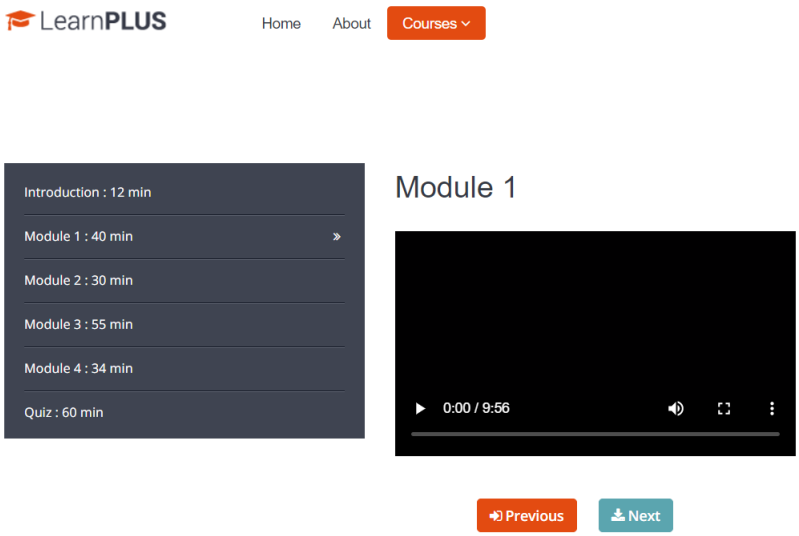



Figure 4.5: Course Content to be used by Learner

### 4.0.2 Back-End

It's worth noting that the front end of the project is just one component of a larger system, which includes a back end for managing the data and a database for storing it. However, the course details page and its associated modules are critical to the user interface and overall user experience. In our project, we have used Express as our back-end technology, and the main file that runs the application is app.js. This file is responsible for setting up the server and defining the routes and middleware that handle incoming requests and responses.Below schemas are used for course data collection.

```
//* Initializing Middlewares
app.use(helmet());
app.use(cors());
app.use(xss());
app.use(mongoSanitize());

app.use(express.json());
app.use(express.urlencoded({ extended: true }));

//* Serving Frontend
// app.use(express.static(path.join(__dirname, 'public')))

//* Serving Docs
app.use('/api-docs', swaggerUI.serve, swaggerUI.setup(swaggerDocuments));

//* API Routes
app.use('/byexpertise', routes);
app.use('/analytics', analyticalRoutes)

//* Error Handling middleware
app.use(notFoundMiddleware, errorHandlerMiddleware);

const start = async () => {
  try {
    await connectDB(api_keys.MONGO_URL).then(() => console.log('Connected to DB..'));
    app.listen(port, () =>
      console.log(`Server is listening on port ${port}...`)
    );
  } catch (error) {
    console.log(error);
  }
};

start();
```

Figure 4.6: App.js file for Express App

```
const mongoose = require('mongoose');
const {Schema} = mongoose;

const courseData = new Schema ({
  courseId: {type: mongoose.Schema.Types.ObjectId, ref: 'courses', required: true},
  pageViews: [{hits: {type: Number, default: 0}, timestamp: {type: String}, _id: false}],
  bounceRates: [{hits: {type: Number, default: 0}, timestamp: {type: String}, _id: false}],
  reviews: [{hits: {type: Number, default: 0}, timestamp: {type: String}, _id: false}],
  courseDescription: [{hits: {type: Number, default: 0}, timestamp: {type: String}, _id: false}],
  courseDetails: [{hits: {type: Number, default: 0}, timestamp: {type: String}, _id: false}],
  Modules: [
    {
      moduleName: {type: String},
      videoSrc: {type: String},
      pause: {type: Object, _id:false},
      seekTime: {type: Object, _id:false},
      play: {type: Object, _id:false},
      rateChange: {type: Object, _id:false},
      endedFirst: {type: Number},
      _id: false
    }
  ]
}, {versionKey: false})
```

Figure 4.7: Course Data Collection Schema

```
const mongoose = require('mongoose');
const { stringify } = require('yamljs');
const {Schema} = mongoose

const UserData = new Schema({
  courseId: {type: mongoose.Schema.Types.ObjectId, ref: 'courses', required: true},
  guestId: {type: String},
  lastUrl: {type: String},
  completed: {type: Number},
  completionPath: [{type:String}],
  timestamp: [{type: String}],
  deviceData: {
    browser: {type: String},
    browserVersion: {type: String},
    platform: {type: String}
  }
}, {versionKey: false})

UserData.pre('save', function (next) {
  if (this.timestamp.length > 100) {
    // Remove oldest timestamp if queue size is exceeded
    this.timestamp.splice(0, this.timestamp.length - 100);
  }
  next();
});

module.exports = mongoose.model('userData', UserData);
```

Figure 4.8: User Data Collection Schema

# Chapter 5

# Testing

Software testing is a vital part of the software development lifecycle that involves checking the software product's quality and functionality. The process aims to identify and report defects, errors, or vulnerabilities in the software before it is released to the end-users. It ensures that the software meets the desired requirements and works as intended.Different types of testing are performed during the software development process, including functional testing, non-functional testing, integration testing, regression testing, and user acceptance testing. Each type of testing has its objectives and procedures.Automated testing tools are commonly used in software testing to increase efficiency and accuracy. These tools can simulate user interactions and run tests continuously, providing faster feedback to the development team.Software testing is crucial in ensuring that the software application is reliable, secure, and user-friendly. It also helps to reduce the risk of software failure or system crashes, which can cause significant damage to the business.In conclusion, software testing is a necessary step in the software development process that helps to identify and fix issues in the software product, ensuring it meets the required quality standards and user expectations.

### 5.0.1 Unit Testing

Unit testing is a software testing technique in which individual units or components of a software application are tested in isolation to ensure that they are functioning as expected. This involves testing each unit of code, such as a function, method, or class, to verify that it behaves as intended and does not produce any errors or unexpected results. Unit tests are typically automated and run frequently as part of a continuous integration and continuous deployment (CI/CD) process, which allows developers to catch defects early in the development cycle and prevent them from becoming more significant issues. Unit testing helps improve the quality of the software, reduce the risk of bugs, and speed up the development process by allowing developers to catch and fix issues quickly.

### 5.0.2 Functional Testing

Functional testing is a type of software testing that verifies whether an application or software system performs its intended functions correctly from an end-user perspective. It involves testing the application's functionality and user interface. The goal of functional testing is to ensure that the application meets the requirements and expectations of the end-users. Test cases are created to simulate typical user interactions with the application to verify that it is providing the expected results. This type of testing is typically performed manually but can be automated. Functional testing is an important part of the software development process as it helps to ensure that the application is delivering the expected functionality and providing a positive user experience. It is often performed in conjunction with other types of testing, such as integration testing and acceptance testing. Overall, functional testing helps to improve the quality and reliability of the software application.

| Test condition | Test Step | Test Data | Expected Result | Actual Result | Pass/Fail |
|---|---|---|---|---|---|
| Element listener | Pass Node element and listener | Check if the element exists in DOM | Event Listener added to the element | Event listener Working | Pass |
| Media listener | Pass media element and listener | Classify listeners | Video data capturing | Data stored in Array | Pass |
| Push User Data | Pass data from the Front-end | Check if the user exists in database | User data gets added | User gets created/ updated | Pass |
| Aggregating Data | Pass raw data | Check for timestamps and already stored documents | Data gets added cumulatively | Documents are updated | Pass |

Table 5.1: Possible Test Cases

# Chapter 6

# Result

By using this analytical tool, we were able to capture valuable data on students' behavior and performance in online courses. We used the metrics defined in the methodology to determine the data requirements. Based on the data collected, instructors can identify areas of the course where students were less engaged and make targeted improvements to increase engagement. For example, a useful metric is course conversion, which is determined by comparing total course page visits against the course enrollment.


Figure 6.1: Page Views vs Students enrolled

In the above figure the blue line denotes the number of students enrolled to the course against the number of page visits represented in pink bar chart. This metric can help instructors understand whether the first impression of course is good enough. We can also monitor the interactions of students with the course material. If the material contains video lectures we can capture multimedia events such as pause, play, skip, etc. An example of Seektime is given below Seektime is the part or time of the video where the students skips to.


Figure 6.2: Bounce Rates

The above figure represents the bounce rate for a particular course, where X-axis represents dates and Y-axis represent total counts of bounce rate for a day. This helps to gain insights about the bounce rate for a particular date and what is the relation between different days to analyse the rise or fall of bounce rate. The X-axis represents the time in seconds

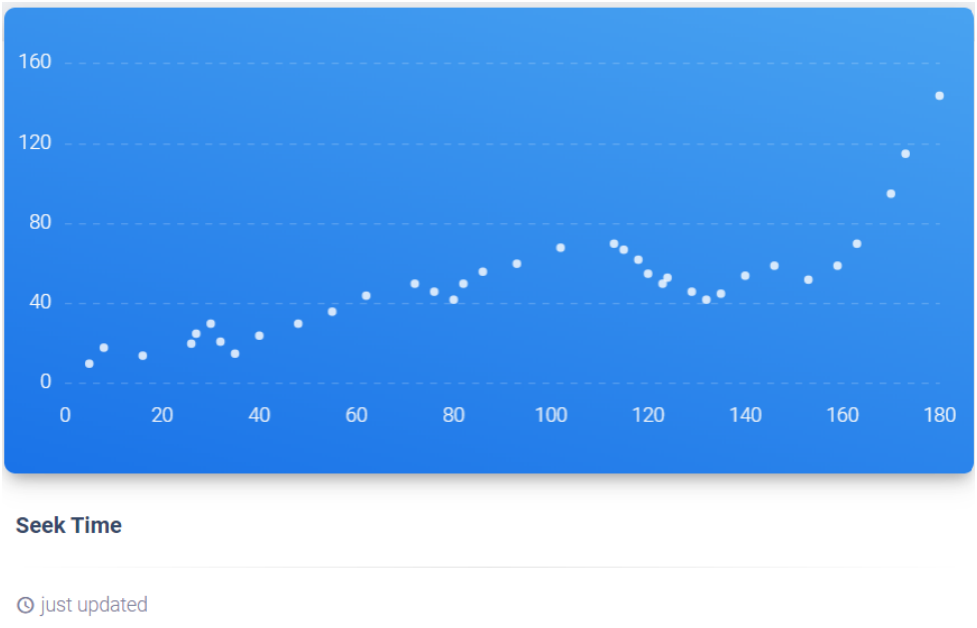



Figure 6.3: Seek Time in Video

concerning the video and Y-axis represents the number of times the video was skipped. This helps to determine whether the students are skimming through the course for the sake of completion. From the above example we can clearly see the video was skipped to the end

many times. We can also check which module is popular among students to complete first in a course. The following figure represents that data using a pie chart.

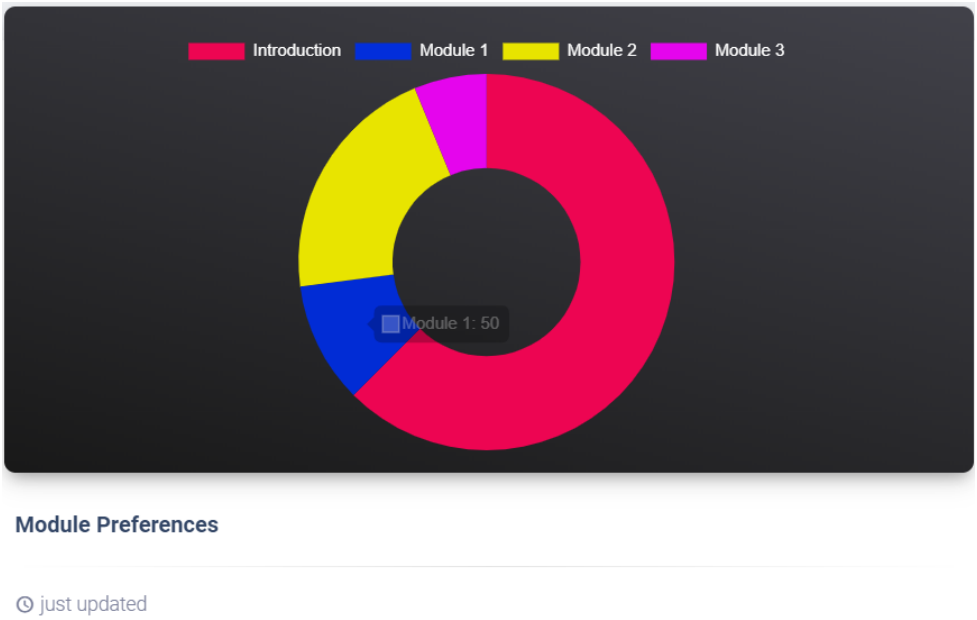



Figure 6.4: Student's Module Preferences

The chart represents the distribution of modules that are completed first in the course. This can help instructors better understand student requirements from the course. For instance, in the above example "Module 1" has the highest rate of completion which means majority of students are familiar with the prerequisites and introductory material

# Chapter 7

# Conclusions and Future Scope

In conclusion, building an analytical tool to capture student interactions with online learning platforms is a critical step in improving the overall learning experience for students. By providing instructors with valuable insights into how student interact with their courses, this tool can help them make data-driven decisions on improving the course content. The development of learning analytics to increase student engagement and performance has the potential to revolutionize the way we approach online learning.

The future scope of this project is vast as the use of online learning platforms is constantly increasing. Some of them are discussed below. Firstly, the data collected can grow exponentially in this kind of application based on the popularity of the course. So decision making for instructors could become difficult. We can implement predictive analysis and reduce the overhead on the end user, showing them filtered and only necessary results. Secondly, we can provide personalized learning experiences for students by tracking their progress and making recommendations for courses that match their learning style.

# Bibliography


[1] Prince, Michael Felder, Richard Brent, Rebecca. (2020). Active student engagement in online stem classes: approaches and recommendations. Advances in Engineering Education. 8.

[2] Heilporn, Géraldine & Lakhal, Sawsen Belisle, Marilou. (2022). Examining effects of instructional strategies on student engagement in blended online courses. Journal of Computer Assisted Learning. 38. 10.1111/jcal.12701.

[3] Leitner, Philipp Maier, Karin Ebner, Martin. (2020). Web Analytics as Extension for a Learning Analytics Dashboard of a Massive Open Online Platform.10.1007/978-3-030-47392-119.

[4] Dhumantarao, Thammi Raju Murthy, G.R.K. Khade, Shrikant Padmaja, B. Bs, Yashavanth Kumar, S. Soam, Sudhir Ch, Srinivasrao. (2021). Understanding Learner Behaviour in Online Courses through Learning Analytics. Asian Journal of Agricultural Extension, Economics Sociology. 381-390. 10.9734/ajaees/2021/v39i1030705.

[5] Gray, Julie Diloreto, Melanie. (2016). The Effects of Student Engagement, Student Satisfaction, and Perceived Learning in Online Learning Environments. International Journal of Educational Leadership Preparation. 11. 89-119.

[6] İ. Yıldırım and S. Çırak-kurt , "The Effect of Gamification on Learner Motivation: A Meta-Analysis Study", Education Reform Journal, vol. 7, no. 1, pp. 1-15, Jun. 2022

[7] X. Solé-Beteta, J. Navarro, B. Gajšek, A. Guadagni, and A. Zaballos, "A Data-Driven Approach to Quantify and Measure Students' Engagement in Synchronous Virtual Learning Environments," Sensors, vol. 22, no. 9, p. 3294, Apr. 2022, doi: 10.3390/s22093294.

[8] Altuwairqi, K., Jarraya, S.K., Allinjawi, A. et al. Student behavior analysis to measure engagement levels in online learning environments. SIViP 15, 1387–1395 (2021).

[9] Czerkawski, B.C., Lyman, E.W. An Instructional Design Framework for Fostering Student Engagement in Online Learning Environments. TechTrends 60, 532–539 (2016).

[10] Yadav, R. K., Singh, A., Arora, M. (2021). A Comprehensive Review of Data Analytics Techniques for Business Intelligence. International Journal of Emerging Technologies in Engineering Research, 8(1), 60-66. https://doi.org/10.30534/ijeter/2021/258112021

[11] Tuti Purwoningsih;Harry B. Santoso;Zainal A. Hasibuan; (2020). Data Analytics of Students' Profiles and Activities in a Full Online Learning Context . 2020 Fifth International Conference on Informatics and Computing (ICIC), (), –. doi:10.1109/icic50835.2020.9288540

# Appendices

## Appendix: Project Setup and installation

1. Download and install NodeJS for your specific machine from https://nodejs.org/en/download And also create an account on https://www.mongodb.com/cloud/atlas/register

2. Download the project zip and extract it in a your local machine.

3. Go to the project and type the following commands in terminal
**npm install**
**npm audit fix**
**npm audit fix –force**

4. Now create a .env file in the project root folder and define the following values.
**URL=Your Base Url**
**MONGO_URL=Your Mongo DB Atlas URL**
**JWT_KEY=Your JWT secret**

5. Setup the scripts for starting the nodemon server by making the following changes to the package.json file.
"scripts": {
"start": "nodemon app.js",
"test": "echo Ërror: no test specified¨ exit 1"
}
Nodemon is helpful since it automatically restarts the server after encountering any changes

6. Now to run the above defined scripts open the terminal and type the following command
**npm start**

7. THe project will run successfully if it shows Connected to DB in the console. Now open the localhost url for port 5000 and interact with the website to generate data on your mongo db atlas project. Enjoy

# Publication

Paper entitled **"Developing Data Analytics Support for Creative Learning Web Framework"** is awaiting decision at **"International Conference on Sustainable Computing and Smart Systems (IEEE - ICSCSS 2023)"** by **"Neel Dudheliya, Anand Morye, Ayush Jain, Prof. Jayshree Jha and Dr. Kiran Deshpande".**