

A DIGITAL ELEARNING EDUCATIONAL TOOL LIBRARY FOR SYNCHRONIZATION  
COMPOSITION & ORCHESTRATION OF LEARNING SESSION DATA

by

ALEXANDROS PAPADAKIS

Informatics Engineer, Technological Education Institute of Crete, 2018

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

SCHOOL OF ENGINEERING

HELLENIC MEDITERRANEAN UNIVERSITY

2022

Approved by:

Major Professor  
Nikolas Vidakis

# **Copyright**

ALEXANDROS PAPADAKIS

2022

## **Abstract**

In the era of the 4th industrial revolution (and especially with the conditions prevailing due to coronavirus pandemic) it became essential mandatory to enhance existing and to develop new digital eLearning educational ecosystems with digital classrooms, digital learning processes, educational games, etc. Because of the increased use of digital eLearning educational tools, new ideas and solutions are constantly sought and better utilization, through various tools for controlling processes, are researched. Moreover, vast amounts of data and information, in various formats and digital repositories is created, making the management and orchestration of all digital media extremely difficult. This results in the evaluation and improvement of learning sessions to remain unexploited. Platforms and ecosystems supporting digital learning conferences such as the IOLAOS platform can play a crucial role for the creation of all this important data. IOLAOS is a platform for learning, based on the contribution and collaboration of various fields of expertise, where teachers and students have an important role through digital learning sessions with modern technologies, tools, or simple video conferencing. Through these sessions, a large amount of data is generated which is stored with the help of various technologies and is available for future use and utilization. This data is of various forms and formats such as: (a) video (student screen recording, session recording, face recording), (b) audio, (c) data from emotion recognition system (image processing), (d) data triplets of educational activities of various events as defined by the xAPI standard (e.g. interactive chat, game, questionnaire, exercise, etc.). In literature, there are various tools which, through the synchronization of several modalities, produce numerous useful data which help either to solve different situations or to improve them. Some scenarios from such tools are the investigation of the truth and the law enforcement of a violent incident by synchronizing many recorded videos and sounds from various cameras, such as surveillance, or smartphone cameras. In addition, synchronizing video from different angles of an important event, as well as mining specific sounds from synchronizing a video collection, are some examples of the large list of similar synchronization tools in several modalities for exporting important data. The amount of data generated is an untapped treasure trove of information that can contribute to the production of many important conclusions that would be extremely difficult or impossible to produce with conventional methods and without the use of digital tools. These conclusions are particularly important as the digital transition of learning creates a new environment in which we

apply principles from other environments due to the lack of understanding of this new technology. The difficulty for the extraction of various and important conclusions lies in the differentiation of the coding of the stored data and the lack of a common line in time. The need to draw these conclusions led us to the idea of implementing a tool for the synchronization - composition - orchestration of all this valuable untapped data. The aim of this tool is to combine a variety of learning data into simple and understandable forms of information that will lead the teacher to a better understanding of the strengths and weaknesses of students, the lesson, the educational process and himself by making a critical look at the above with aimed at a substantial improvement of all components of the learning path. In order to be able to synchronize the data from different knowledge repositories we must have access to what requires the use (where available) or the development of Restful APIs as well as the implementation of data space to store data from the above sources and to provide access to the information produced. In addition, you will need a front end for data orchestration which will be developed with new web application technologies such as React JS, Node JS.

## Περίληψη

Στην εποχή της 4ης βιομηχανικής επανάστασης (και ειδικά με τις συνθήκες που επικρατούν λόγω της πανδημίας του κορωνοϊού) κατέστη απαραίτητο να ενισχυθούν τόσο τα υπάρχοντα όσο και να αναπτυχθούν νέα ψηφιακά εκπαιδευτικά οικοσυστήματα eLearning με ψηφιακές αίθουσες διδασκαλίας, ψηφιακές διαδικασίες μάθησης, εκπαιδευτικά παιχνίδια κ.λπ. Λόγω της αυξημένης χρήσης των ψηφιακών εκπαιδευτικών εργαλείων eLearning, αναζητούνται συνεχώς νέες ιδέες και λύσεις και ερευνάται καλύτερη αξιοποίηση, μέσω διαφόρων εργαλείων ελέγχου των διαδικασιών. Επιπλέον, δημιουργούνται τεράστιοι όγκοι δεδομένων και πληροφοριών σε διάφορες μορφές, καθώς και ψηφιακά αποθετήρια, καθιστώντας εξαιρετικά δύσκολη τη διαχείριση και την εντοπισμένη χρήση όλων των ψηφιακών μέσων. Αυτό έχει ως αποτέλεσμα η αξιολόγηση και η βελτίωση των συνεδριών μάθησης να παραμένουν ανεκμετάλλευτες. Οι πλατφόρμες και τα οικοσυστήματα που υποστηρίζουν συνέδρια ψηφιακής μάθησης, όπως η πλατφόρμα IOLAOS, μπορούν να παίξουν καθοριστικό ρόλο για τη δημιουργία όλων αυτών των σημαντικών δεδομένων. Ο IOLAOS είναι μια πλατφόρμα μάθησης, βασισμένη στη συμβολή και τη συνεργασία διαφόρων τομέων εξειδίκευσης, όπου δάσκαλοι και μαθητές διαδραματίζουν σημαντικό ρόλο μέσω ψηφιακών συνεδριών μάθησης με σύγχρονες τεχνολογίες, εργαλεία ή μέσω απλής τηλεδιάσκεψης. Μέσα από αυτές τις συνεδρίες δημιουργείται μεγάλος όγκος δεδομένων ο οποίος αποθηκεύεται με τη βοήθεια διαφόρων τεχνολογιών και είναι διαθέσιμα για μελλοντική χρήση και αξιοποίηση. Αυτά τα δεδομένα είναι διαφόρων τύπων και μορφών όπως: (α) βίντεο (εγγραφή οθόνης μαθητή, εγγραφή συνεδρίας, εγγραφή προσώπου), (β) ήχος, (γ) δεδομένα από σύστημα αναγνώρισης συναισθημάτων (επεξεργασία εικόνας), (δ) τριπλέτες δεδομένων εκπαιδευτικών δραστηριοτήτων διαφόρων εκδηλώσεων όπως ορίζονται από το πρότυπο xAPI (π.χ. διαδραστική συνομιλία, παιχνίδι, ερωτηματολόγιο, άσκηση κ.λπ.). Στη βιβλιογραφία, υπάρχουν διάφορα εργαλεία τα οποία, μέσω του συγχρονισμού πολλών τρόπων, παράγουν πολυάριθμα χρήσιμα δεδομένα που βοηθούν είτε στην επίλυση διαφορετικών καταστάσεων είτε στη βελτίωσή τους. Μερικά σενάρια από τέτοια εργαλεία είναι η διερεύνηση της αλήθειας και η επιβολή του νόμου ενός βίαιου περιστατικού με συγχρονισμό πολλών εγγεγραμμένων βίντεο και ήχων από διάφορες κάμερες, όπως κάμερες παρακολούθησης ή smartphone. Επιπλέον, ο συγχρονισμός βίντεο από διαφορετικές οπτικές γωνίες ενός σημαντικού

γεγονότος, καθώς και η εξόρυξη συγκεκριμένων ήχων από το συγχρονισμό μιας συλλογής βίντεο, είναι μερικά παραδείγματα της μεγάλης λίστας παρόμοιων εργαλείων συγχρονισμού σε διάφορους τρόπους εξαγωγής σημαντικών δεδομένων. Ο όγκος των δεδομένων που δημιουργείται είναι ένας αναξιοποίητος θησαυρός πληροφοριών που μπορεί να συμβάλει στην παραγωγή πολλών σημαντικών συμπερασμάτων που θα ήταν εξαιρετικά δύσκολο ή αδύνατο να παραχθούν με συμβατικές μεθόδους και χωρίς τη χρήση ψηφιακών εργαλείων. Αυτά τα συμπεράσματα είναι ιδιαίτερα σημαντικά καθώς η ψηφιακή μετάβαση της μάθησης δημιουργεί ένα νέο περιβάλλον στο οποίο εφαρμόζουμε αρχές από άλλα περιβάλλοντα λόγω της έλλειψης κατανόησης αυτής της νέας τεχνολογίας. Η δυσκολία για την εξαγωγή ποικίλων και σημαντικών συμπερασμάτων έγκειται στη διαφοροποίηση της κωδικοποίησης των αποθηκευμένων δεδομένων και στην έλλειψη μιας κοινής γραμμής στο χρόνο. Η ανάγκη εξαγωγής αυτών των συμπερασμάτων μας οδήγησε στην ιδέα της εφαρμογής ενός εργαλείου συγχρονισμού – σύνθεσης – ενορχήστρωσης όλων αυτών των πολύτιμων αναξιοποίητων δεδομένων. Στόχος αυτού του εργαλείου είναι να συνδυάσει μια ποικιλία μαθησιακών δεδομένων σε απλές και κατανοητές μορφές πληροφοριών που θα οδηγήσουν τον δάσκαλο στην καλύτερη κατανόηση των δυνατών και αδυναμιών των μαθητών, του μαθήματος, της εκπαιδευτικής διαδικασίας και του εαυτού του, κάνοντας μια κριτική ματιά. στα παραπάνω με στόχο την ουσιαστική βελτίωση όλων των συνιστωσών της μαθησιακής διαδρομής. Για να μπορέσουμε να συγχρονίσουμε τα δεδομένα από διαφορετικά αποθετήρια γνώσης πρέπει να έχουμε πρόσβαση σε ό,τι απαιτεί τη χρήση (όπου είναι διαθέσιμο) ή την ανάπτυξη Restful API καθώς και την υλοποίηση χώρου δεδομένων για την αποθήκευση δεδομένων από τις παραπάνω πηγές και την παροχή πρόσβαση στις πληροφορίες που παράγονται. Επιπλέον, θα χρειαστεί ένα γραφικό περιβάλλον για την ενορχήστρωση δεδομένων που θα αναπτυχθεί με νέες τεχνολογίες διαδικτυακών εφαρμογών όπως React JS, Node JS.

# Table of Contents

Copyright .....	ii
Abstract .....	iii
Περίληψη .....	v
Table of Contents .....	vii
List of Figures .....	ix
List of Tables .....	x
Acknowledgements .....	xi
Chapter 1 - Introduction .....	1
Chapter 2 - Background .....	4
2.1 Data Orchestration .....	4
2.1.1 Current technological trends in the steps of data orchestration .....	4
2.1.2 Orchestration in learning .....	5
2.2 Video & Audio Synchronization .....	6
2.2.1 Techniques .....	6
2.3 Data Visualization .....	7
2.4 Learning Analytics: the Experience API .....	9
Chapter 3 - The Arion System .....	11
Arion Software Analysis .....	12
Functional and Non-Functional Requirements .....	12
Use Cases .....	18
Arion Software Architecture .....	21
Chapter 4 - Implementation .....	23
Frameworks – Tools & Platforms used .....	23
IOLAOS .....	23
Emotion API .....	24
Technologies Used .....	26
MongoDB .....	26
NodeJS .....	27

ReactJS.....	29
REST API & Databases .....	30
Chapter 6 - Pilot Use Case .....	32
Chapter 7 - Conclusion & Future Work.....	40
References Or Bibliography .....	41



## List of Figures

Figure 2-1 bar chart.....	8
Figure 2-2 pie chart.....	8
Figure 2-3 Histogram.....	8
Figure 2-4 scatter plot.....	9
Figure 2-5 scatter plot 3D.....	9
Figure 2-6 Network analysis.....	9
Figure 2-7 Gantt chart.....	9
Figure 2-8 Heat map.....	9
Figure 2-9 Box and Whisker plot.....	9
Figure 2-10 - IOLAOS Architecture Overview [26].....	24
Figure 3-1 - Arion System Overview.....	12
Figure 3-2 - External Functional Requirements.....	13
Figure 3-3 - General Functional Requirements.....	15
Figure 3-4 - Modalities Functional Requirements.....	16
Figure 3-5 - Timebar Functional Requirements.....	16
Figure 3-6 - Notes Functional requirements.....	17
Figure 3-7 - Login to Arion Studio Use Case.....	19
Figure 3-8 - Timebar Use Case.....	20
Figure 3-9 - Notes Use Case.....	20
Figure 3-10 - Arion Component Diagram.....	22
Figure 4-1 Node.js Architecture.....	28
Figure 4-2 - Generic RESTful API Architecture.....	31
Figure 5-1. First Figure in Appendix A.....	<b>Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.</b>

## List of Tables

Table 2-1 Techniques of Data Visualization .....	8
Table 3-1 Functional Requirements .....	14
Table 3-2 - Non-Functional Requirements .....	17

## **Acknowledgements**

I would like to express my sincere gratitude to Dr. Nikolaos Vidakis, my supervisor, for his valuable guidance, trust and understanding during the long procedure of writing this thesis. I would also like to warmly thank all team members from NILE Lab for their help and collaboration. Lastly, I thank my family and close friends for providing perspective, encouragement and constant moral support.

## **Chapter 1 - Introduction**

The past couple of decades are characterized by the exponential economic and technological growth. During the first quarter of the 21<sup>st</sup> century we have come to witness unforeseen disruptions in all aspects of life, especially in the industrial and business spheres. The catalyst behind this accelerated evolution is the increased processing capabilities of modern electronics in conjunction with the internet and the World Wide Web. These achievements have transformed most aspects of daily life, including communication, education, and entertainment. The recent global reality, due to the unfortunate COVID-19 pandemic, proves how deep technology has got into our lives, and how fast it can respond to new needs. In the matter of days platforms were ready to support the global economy in a remote working situation, learning and education was moved to a virtual classroom for the first time and virtual meetings became the norm, that is still followed in most situations. Humanity was unprepared, but capable to follow this abrupt change in lifestyle and quickly embraced technology as part of most daily activities. Online platforms and trade grew at rates never seen before, while people of all ages turned to digital services for all their needs. In the case of education, researchers have been interested in the benefits technology can offer since the inception of computers, the internet, and all related technologies. If all this prior work was not available during the recent crisis, it would surely be impossible to move most of the global educational systems to a virtual environment. The entirety of techniques, tools and approaches that are related to online and digital ways of learning are summarized in the term eLearning. There are multiple forms of eLearning, including Massive Open Online Courses (MOOCs), Serious Games, Learning Management Systems (LMSs) and more. Despite the differences between types of eLearning, but even between specific platforms, they all share common principles and, in some cases, techniques. To name a few, in many cases cameras are used to stream video of the participants, written conversation mechanics (chats and forums) are in place for participants to converse, quizzes or puzzles are presented to participants while engaging with the lesson, and much more. In all cases, the students have an active role, different from the one in a classroom, even when they are watching a teacher (live or recorded), they have access to a multitude of tools which might engage or distract them from the active subject. Of course, distractions are present in the classroom as well, however one of the differences is that in a digital environment action, or

even inactivity, can be tracked in a different way, unlocking valuable information about the way student learn and interact during their learning sessions. With the advancement in artificial intelligence and image processing it is possible to identify not only faces, but even the emotions of a person just by a video feed. With learning analytics, it has become easy to understand when, where and how students learn the most, as well as what parts of the lesson they don't focus on. Through live chats and chat bots it is possible to extract valuable information, or even motivate students that are facing difficulties at any given moment. It is obvious that learning has radically changed, offering a multitude of tools to teachers.

Serious Games, introduced as early as 1970 [1], are maybe the most complex and sophisticated, in the technical scope, of the tools used in education. Serious, or Educational, Games are games that have a decisive educational goal, and lean heavily on it, leaving entertainment as a secondary goal of the game [2], [3]. In the early years there was heavy criticism towards serious games, but this is now in the past, with scientist and educators endorsing games as valuable educational tools, with very high potential for education [4], [5]. The success of educational games as tools have spread to the point where educators wish to create their own small games, without having the technical skills or background. Thus, a set of tools, Authoring Tools, were created, which educators can use to create simple games, puzzles or interactive stories with which their students can have an interactive experience while learning [6].

Furthermore, there is a growing interest in adaptive and personalized learning [7]. In adaptive learning techniques are used to adjust the way each and every student is presented with educational material, affecting the presentation or navigation for instance, and thus creating a personalized syllabus for each student, ultimately leading to all students reaching the same level of knowledge, but through different learning paths [8].

Modern educational tools have expanded to rich and multiple features with complex functionalities. Each of those features play a significant role in the learning process is yet to be understood, and it is crucial to improving digital, and traditional, learning [4]. The current state of eLearning is highly fragmented, which in many cases is a very positive reality, offering diversity and multiple capabilities, but at the same time it is almost impossible to study the effects and deeply understand the benefits and drawbacks and how to improve each and every approach. Based on this observation, we wish to utilize the logging systems available in modern eLearning tools to assist educators to better monitor their learning sessions and extract valuable conclusions

about the procedure, the way their students learn, and the tools used. In later works we aim to automate this assessment procedure so that it can be applied to big data collections and unlock valuable knowledge about learning in the digital world.

## Chapter 2 - Background

### 2.1 Data Orchestration

Data Orchestration in software engineering is linked mostly with service-oriented programming, virtualization, and cloud computing. A definition of orchestration in this context is the coordination and management of various entities to achieve the best possible outcome [9]. Depending on the context, orchestration can be further defined to better describe its purpose in the specific context. In service-oriented programming, the term can be defined as “a combination of services to create higher level services and processes” [10]. In the field of computer-supported collaborative learning (CSCL) it is defined as: “the process of productively coordinating supportive interventions across multiple learning activities occurring at multiple social levels” [11]. The amount of data produced on an organizational, or even applicational, level is increasing daily at high rates. For that reason, the technologies used for storage management systems changing frequently to best cover the current needs. One can say if the data is managed in a good way there is not the need for data orchestration, but this is a hard task to accomplish especially for legacy applications, already existing and producing data for years, creation data separation and through multiple management systems. To unify data management and research between all systems data orchestration is needed. Data orchestration solves problems as deduplication, lead-to-account matching, and data cleansing. In general, it facilitates the minimization of missed data and ensures data integrity. Furthermore, by using data orchestration privacy laws can be followed with ease, guarantee the compliance with them, and provide the necessary information they require (how, when and where the user data are used).

#### *2.1.1 Current technological trends in the steps of data orchestration*

Any data orchestration tool or platform has a goal to highlight data overlooked from previous procedures applied to the provided set. Thus, each data point is allowed to act in harmony with all other available data, while it is also examined as an individual unit and as a part of the larger data set. Due to the wide spectrum of applications and fields utilizing these techniques, data orchestration solutions present variety in their workflows, procedures, and milestones. There is no universal system, however we can identify a general association of procedures and workflows that can define data orchestration. This workflow is comprised of the following steps [12]

**Organize:**

In this step, data is shaped and structured according to the system before admission to ensure integrity and formatting. This step includes data validation to ensure that the incoming data are correct and not malicious, put labels or combine new data with existing information. Usually, APIs are provided from companies for this task to facilitate the process

**Transform:**

After the collection and validation of the data, data orchestration tools transform the data into one standard format. The process of making all the data having the same format makes the data analysis quicker and helps to avoid possible errors, thus minimizing the risk of feeding the system with falsely information that will lead into mistaken conclusions.

**Activate:**

Activation is the process of sending the available data to the tools requesting them. Such tools can be analytics platforms, management solutions, business intelligence and more

***2.1.2 Orchestration in learning.***

Orchestration in learning can be used in various aspects [13]. One of them includes the **planning of the learning activities** in such a way that will ensure that the activities will be achieved. Another aspect is the **management of the learning processes** to maximize the outcome [14]. Furthermore, orchestration can be characterized as the adaptation or intervention into the **design of the learning activities** [15]. Assessment can provide information about how good or bad the learning outcomes are, thus orchestrating the assessment can be as beneficial to learners because the learning material can be adapted to their needs as to the teachers that can better adapt their teaching to the learner [14]. Finally, orchestration can be made from various perspectives, usually it is from the teacher perspective but there are approaches that takes the perspective of the learner. Orchestration in learning should be used to ensure that combined learning activities can be well-orchestrated to deliver the desired learning outcome despite conditions that might change during the learning process [11]. Robust orchestration models for various scenarios of learning should be created as the complexity of learning is increasing [13].



## **2.2 Video & Audio Synchronization**

Video and audio synchronization is the attempt to gather information about an event from various devices or time segments and organize this information in correct order so that the produced outcome will reflect the truth about the event [16]. Nowadays, the need for audio and video synchronization has peaked, due to the wide-spread use of mobile devices with embedded cameras, resulting in multiple recordings of the same event, from multiple attendees, without the need of specialized equipment. This results to videos and audios with low resolution, blurry, and most often noisy [16], [17]. Having much data about an event from different sources makes the extraction of information hard, when that data is also with different quality or specifications, the extraction task is even harder. Therefore, synchronization of video and audio have many applications in the modern world, from free-view video to the detection of human rights violations [16].

### ***2.2.1 Techniques***

Most common techniques to synchronize two or more videos are based on tracking visual features [18], [19]. Such methods require the same visual features to be visible in both videos, something that is not always applicable as videos can be captured from a different perspective or location. Another technique is to mark the beginning of videos using clapping boards or by using jam sync which synchronizes the camera clocks [20], but these techniques are not applicable “in the wild”, as both are professional techniques requiring not only expensive hardware, but also a stable set up of cameras in a controllable environment.

Due to the limitations of the visual features tracking, audio synchronization is used for outdoor videos. Audio fingerprinting has proven a promising technique for synchronization. This in turn imposes new challenges, as microphones capturing the scene might be far apart or the sound might cut on one of them. Furthermore, sound can lose quality from compression or noise, thus leading to deviations in pairwise matches. RANSAC or similar methods are regularly used to improve and form a more robust result.

Bottom-up approaches are the most used according to the literature [19], [21]. This approach starts by matching single pairs and continues by merging them gradually until it reaches a global alignment. An implementation of this perspective is using audio fingerprinting to match single pairs into larger clusters through reversed-indexing correlation evaluation. There are many similar

techniques, as in [22], that applies clustering techniques to the matched scores. However, bottom-up approaches have proven frailty to outliers, as a result of poor matches which should be removed from the start, otherwise the false data can be gathered throughout the process resulting into erroneous outcomes.

## **2.3 Data Visualization**

Data visualization is a research field that studies how data can be visualized to be better understood by humans. It is known that for the human eye it is easier to spot differences in shapes, for instance what line is the longest, or what colors are included in a canvas, rather than in a table of numbers. Therefore, data visualization can be traced back in time to the beginning of civilization, where tables with the position of the stars and other celestial objects were used. Geometric diagrams have also been used for many centuries as well as exploration and navigation maps [23]. In 17<sup>th</sup> century coordinates systems, analytic geometry, and theories of errors of measurement (Descartes, Fermat, Galileo) made their initial steps. Later in the 18<sup>th</sup> century, with the help of statistics, more information started to appear on maps as economics. At the start of the 19<sup>th</sup> century the modern graphics were established with the first histograms, bars, pie charts and more. During the same century the first 3-D surface plot appeared. The next century started with limited advancements only to give space on the second half that the leaf plots and boxplots materialized, better organization of data established based on visual and perceptual elements, and finally the FORTRAN programming language was developed, which was the first high level programming language. From 1975 and after the data visualization is a complete research area.

Many disciplines are contributing into data visualization [24]; one is psychology that studies the impact of shapes, colors, sizes, and the perception of data. Computer science and statistics help to develop new techniques and technologies to handle data, such as machine learning and data mining. Additionally, infographics and dashboards are created using graphical and multimedia techniques using different shapes, colors, scales, and data.

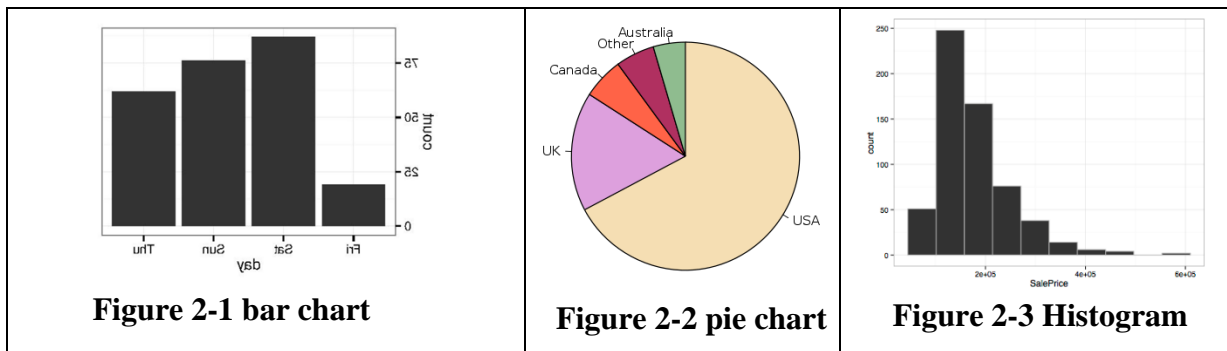
Through the graphical representation of data, it is easier for the human eye to interpret the information provided to them. There are many techniques and diagrams to better suit the data that needs to be displayed. As stated before, shapes colors and sizes can help humans to understand the data, but the overuse of colors might be misleading, so it needs careful treatment to ensure the

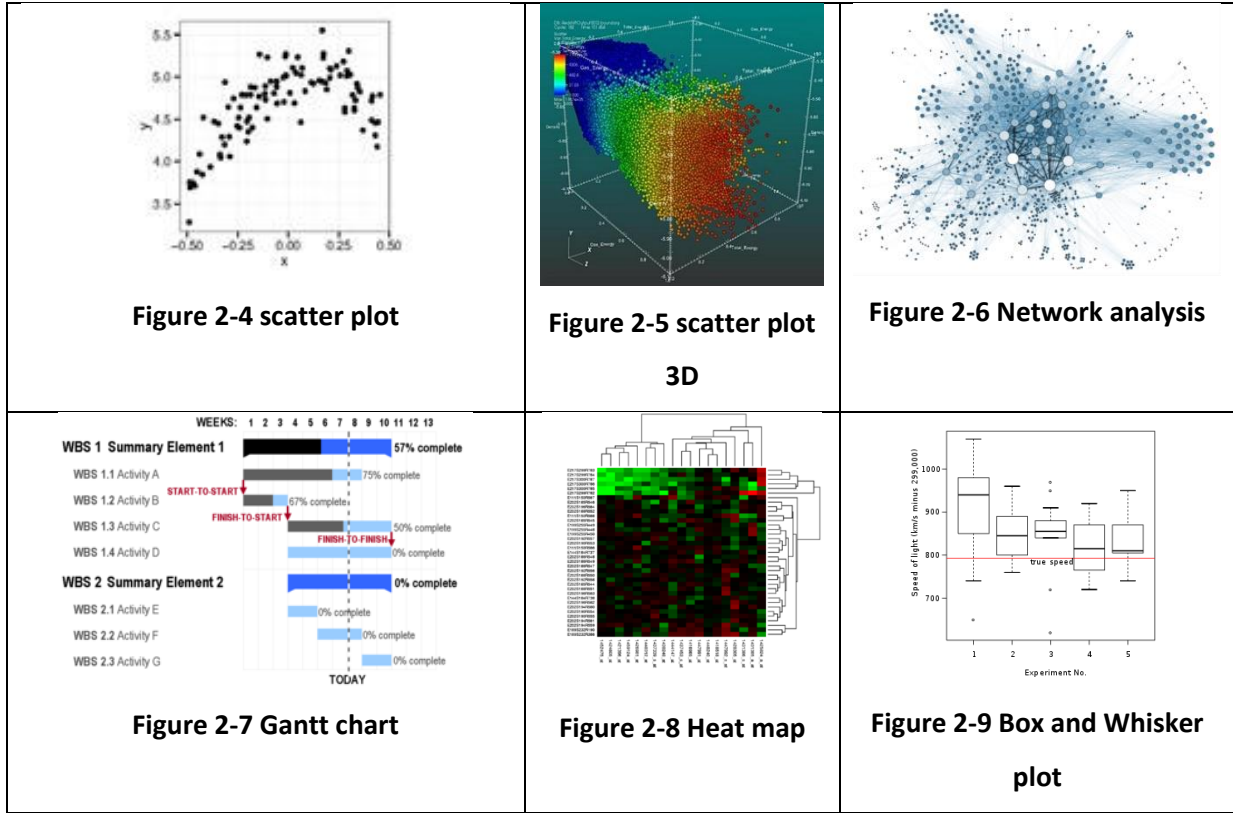
correct meaning is communicated. Another aspect that needs attention is the highlight of unimportant information that can destruct from the important data on the diagrams [24].

As data visualization is widely used nowadays, from marketing and economy to demographics, an increasing number of people are involved with this field. That leads to new challenges that needs further study [25]. The first major challenge is that people that need to use or produce data visualizations can have very varied backgrounds, and thus there is a need for tools that will make it possible for people to create and understand data visualization from heterogeneous datasets that might contain noise or bad quality entities, without the need for a strong data science or IT background. Additionally, there is a need for tools that will allow data scientists to perform data exploration without the requirement of data manipulation or analytics skills. Furthermore, as the data are increasing, a necessity for new techniques to provide fast results has arisen. For closure, with the advancements of artificial intelligence and machine learning, research on how to capitalize on those techniques to provide better graphs depending on the content of the data is promising.

A cheat sheet for data visualization techniques is presented by Wang et. al [26], while some of the most common techniques for data visualization are presented in Table 2-1 Techniques of Data Visualization:

**Table 2-1 Techniques of Data Visualization**





## 2.4 Learning Analytics: The Experience API

Learning analytics is a recently developed discipline that focuses on using sophisticated analytic techniques in a learning environment with a goal to improve education. It draws on a variety of other fields of study, including game and web analytics, business intelligence, web educational data mining, and is intimately linked to them. Learning analytics are based on a premise, that learning is a product of interaction [27]. The idea behind this premise is to take a note of each interaction, along with any usable information about it and then analyze the sum of interactions to produce information that can be used to evaluate learning.

Experience API (xAPI) is a framework that implements a standard for learning analytics. Though it, we facilitate data capturing, retrieval, and creation of visual presentation of valuable information, utilizing data visualization techniques. It is used in learning technologies as a middleware for communication between software and Learning Record Store (LRS), which is the database where each action is recorded. This way xAPI enables very different systems to have a secure communication while capturing and sharing data of the experience a user has [26]. The strength of Experience API lies in its flexibility, which stems from its architecture. This

architecture is comprised of statements, that represent a single action, this action is send to an LRS in JSON form through HTTP communications, and retrieved from it when need be, using the same standard. Statements are in the form of “subject-verb-object”, or as the xAPI terms are, “actor-verb-object”, or, in layman’s terms, “Who-Did-What”. This structure gives the possibility of describing any situation imaginable during the creation of learning experiences. Verbs and object can be retrieved from an official xAPI vocabulary, or created by anyone to fulfill current needs, although the available vocabulary is quite extensive and can cover most possible situations.

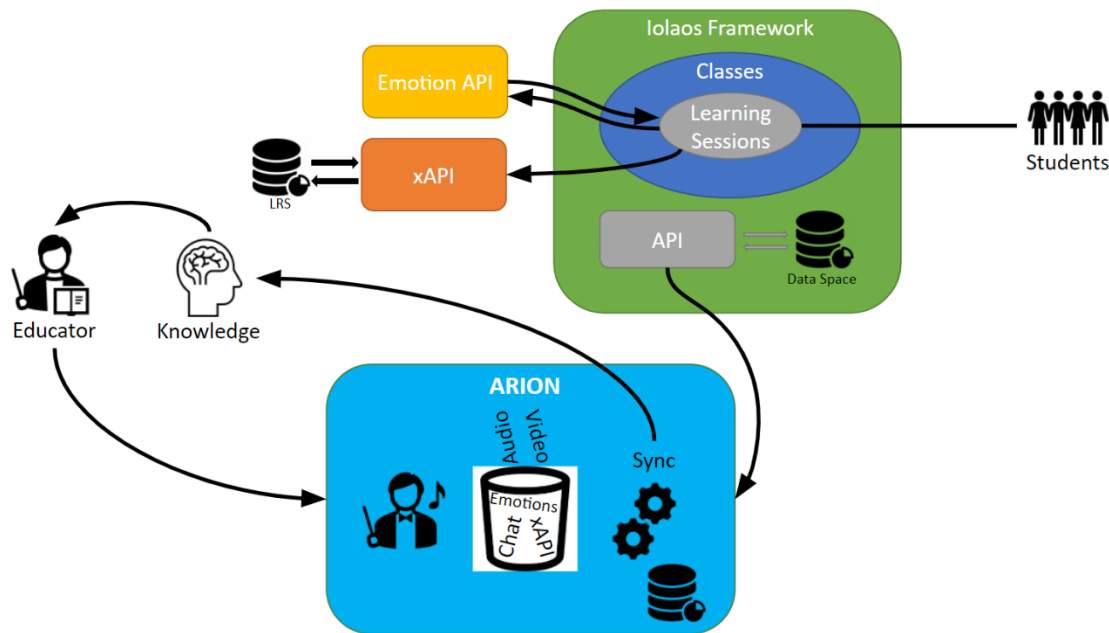
Additionally, IOLAOS has included a maturity level in which the inclusion of xAPI in educational activities is required. For IOALOS, the techniques provided by learning analytics, and xAPI in particular, are crucial in extracting insightful knowledge and understanding about the digital learning procedure. The entirety of tools provided are useful to game developers, teachers and educational experts, and thus is considered a necessity and a discrete maturity level was created. Therefore, we also consider it crucial for analysis and orchestration in our system.

## Chapter 3 - The Arion System

After our research we concluded that a system to unify logging data from multiple sources is necessary to identify weak and strong points in platforms, as well as the specific needs of students. The highly scattered nature of this logging data, as well as the unimaginable numbers of formats and different representations makes this a very challenging task. There are video feeds, propriety and open-source action/logging formats, differing representations of written communications, sound communications and many more forms of data. All this data hides very useful information for the educators, who unfortunately are not able to sift and browse through this unstructured and scattered set of very specific details. For this reason, we have designed a system that will retrieve all data related to a specific learning session, wherever they may be stored and regardless of format, and present them in a common timeline which educators can study and reach insights that they would be unable to if the different sources were not combined. To achieve this, a necessary step is to define and use learning sessions in a technical manner. In this step we adopted the theoretical and technical framework create by Vidakis and Charitakis [28], in which learning sessions are thoroughly examined. Furthermore, utilizing the API created by the aforementioned study, along with complementary systems, such as ExperienceAPI[29] and EmotionAPI [30].

A simplified view of the architecture is illustrated in Figure 3-1 - Arion System Overview. On the one end (top left corner) we have the students, who interact with eLearning systems through the IOLAOS Framework. This can include virtual classrooms and serious games in the current state of IOLAOS but can be expanded and populated with other technologies in the future. Any such change won't affect Arion, as long as the logging systems remain the same. During this interaction login data is created by the actions students take within the systems used. If a serious game is played in a virtual classroom for instance, the game will be logging all interactions in a Learning Record Store, using xAPI, while the screen of the student and the video feed from her web camera will be stored in the IOLAOS Common Data Space. After the session is concluded, the Emotion API will query the Common Data Space and analyze all new videos, creating data about the emotions students had for each second during play. At some other point, the educator has the time to review the results of this learning session, thus taking some time to visit Arion and sync the data for each of his students. Then the Arion will query the IOLAOS API and retrieve any data that exist for the specified learning session and student, to present them to the educator.

Using the different modalities, synced to the same timeline, the educator can make connections such as “At 01:30 student a lost some points due to an error, which made him feel angry, but at 02:00 he made his best score, probably motivated by his previous anger”. Thus, the interaction of the educator with the system has brought knowledge and comprehension of students and procedures that were currently unknown.



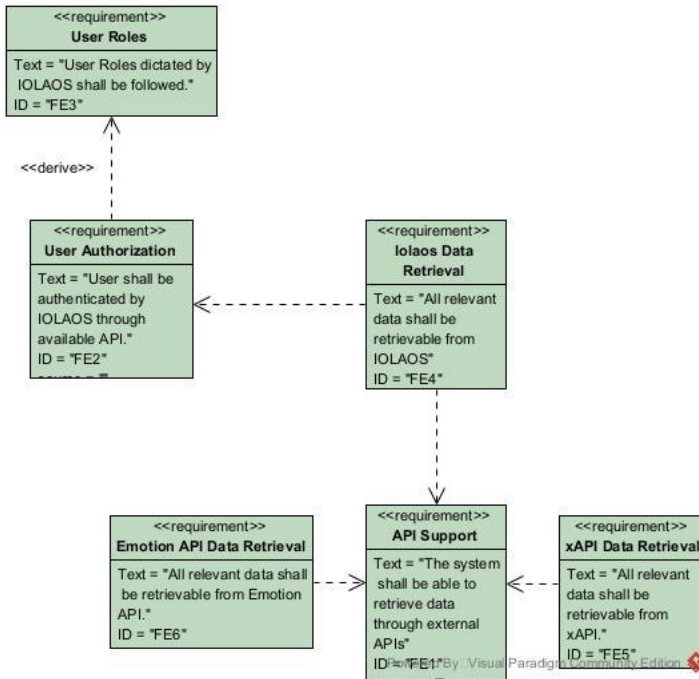
**Figure 3-1 - Arion System Overview**

## Arion Software Analysis

### *Functional and Non-Functional Requirements*

The requirements of the system are extracted from the definition of the problem, as stated above. We created two tables, Table 3-1 Functional Requirements and Table 3-2 - Non-Functional Requirements, categorizing our requirements into the two categories widely used in Software Engineering. We did further categorization by identifying requirements that address similar aspects of the system. For each of the subgroups we created a requirement diagram. Our first category, seen in Figure 3-2 - External Functional Requirements, is the External Functional requirements, which refer to requirements set by the need to collaborate with external tools. So In this category we identify the need to be able to retrieve data through external APIs, which is the

way we will gather the required scattered data of learning sessions. Additionally, we see more specific guidelines that have to do with specific systems we will closely collaborate with, such as authenticating users through the IOLAOS API, following the dictations of IOLAOS regarding user roles, being able to retrieve data from IOLAOS, EmotionAPI and ExperienceAPI.



**Figure 3-2 - External Functional Requirements**

Secondly, we have the Figure 3-3 - General Functional Requirements, a category, in which the general overview of our platform is expressed. In this list we define the need for a web app that will host the Data Synchronization tool, where a list of educator’s classes and sessions will be presented to them, and they will be able to select which to work with at the moment. Additionally, data retrieved will not be editable, to ensure data integrity, while this tool will allow educators to sync and view different modalities in the same timeline.



**Table 3-1 Functional Requirements**

Functional Requirements		
ID	Title	Description
FE <sup>1</sup> 1	API Support	The system shall be able to retrieve data through external APIs
FE2	User Auth	User shall be authenticated by IOLAOS through available API.
FE3	User Roles	User Roles dictated by IOLAOS shall be followed.
FE4	Iolaos Data Retrieval	All relevant data shall be retrievable from IOLAOS.
FE5	xAPI Data Retrieval	All relevant data shall be retrievable from xAPI.
FE6	Emotion API Data Retrieval	All relevant data shall be retrievable from Emotion API.
FG <sup>2</sup> 1	Web App	A web app shall be created to host the Data Synchronization.
FG2	Classes	A list of educator’s classes is presented, from which he/she can choose which to work with.
FG3	Learning Session	Any existing learning session where responsible is the educator, shall be viewable through the app along with data such as number of participants and available media.
FG4	Studio	The studio will allow to sync and view specific modalities.
FG5	Data Editing	Data retrieved shall not be editable.
FM <sup>3</sup> 1	Modality Types	Modality types shall include Text, Video, xAPI Information, extracted Emotion from Image Processing Algorithms.
FM2	Modalities Selection	Educators can select which modality types will be included to the studio, for data synchronization.
FM3	Pupil Selection	Educator can select which pupils and what modalities of them shall be included in the synchronization.

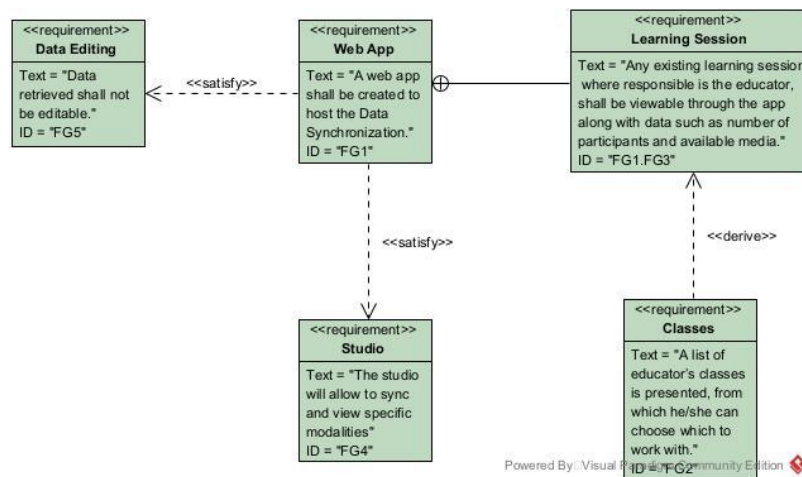
<sup>1</sup> FE = Functional Requirements regarding External API

<sup>2</sup> FG = Functional Requirements regarding General

<sup>3</sup> FM = Functional Requirements regarding Modalities

FM4	Modality Availability	Tracked modalities is accessible only by the educator.
FT <sup>4</sup> 1	Timebar	Timebar shall be available to the user so that time can be scrolled through and access any time of the learning session.
FT2	Timebar Functionality	Timebar shall be configured so that time will be adjusted to the available modalities from the learning session.
FT3	Play/Pause	User can either use the timebar to jump to specific timestamp or use play/pause functionality and watch a rewind of the entire learning session.
FN <sup>5</sup> 1	Note Creation	The educator can create notes at any time of the learning session.
FN2	Note Stored	Note shall be stored in remote servers.
FN3	Note Retrieval	All notes shall be retrievable by the educator.

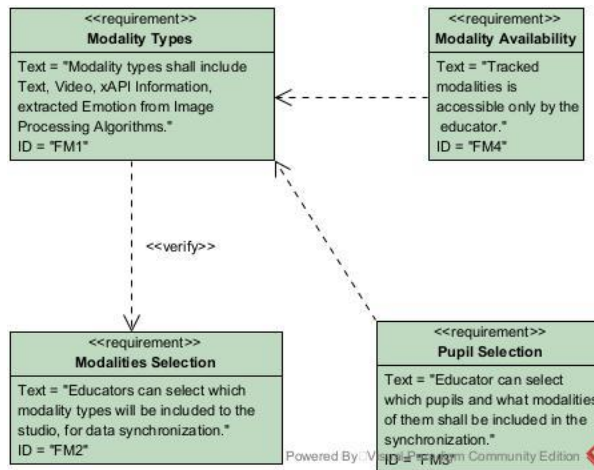
Thirdly, we have Figure 3-4 - Modalities Functional Requirements, which expresses requirements for the types of media and data that the system must support and how they will be handled by the user. This includes definition like the need to support Video, Text, xAPI and EmotionAPI formats, that these modalities will only be accessible to educators of each student, and that educators will have the ability to select which of the available modalities, and for which students, they will be presented with.



**Figure 3-3 - General Functional Requirements**

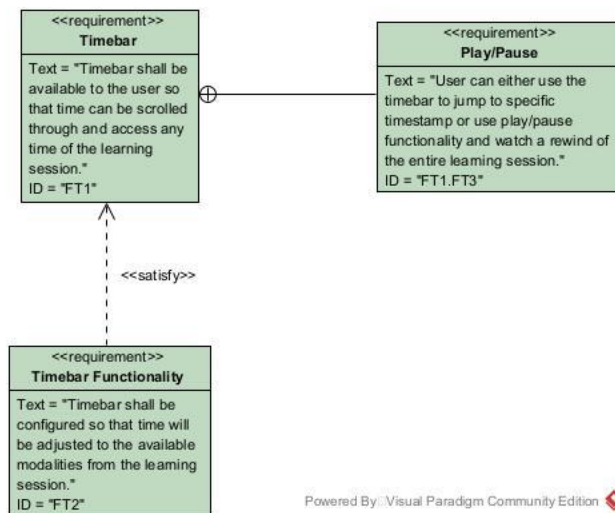
<sup>4</sup> FT = Functional Requirements regarding Timebar

<sup>5</sup> FS = Functional Requirements regarding Note



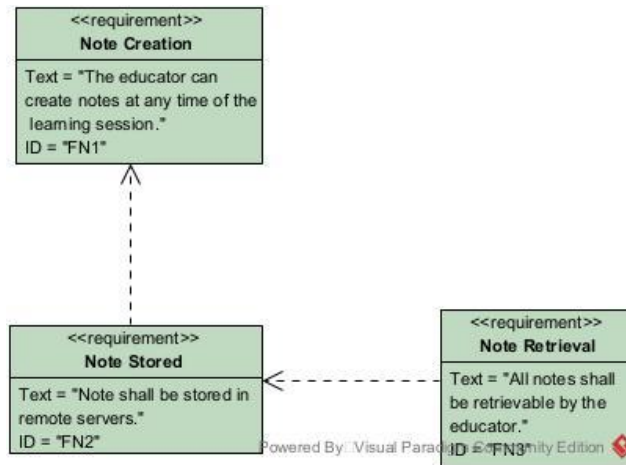
**Figure 3-4 - Modalities Functional Requirements**

Forth, is the Figure 3-5 - Timebar Functional Requirements, which define the way the available Timebar will handle time and how it will interact with the user. As part of those definitions we have three very important points, (a) Timebar shall be available to the user so that time can be scrolled through and access any time of the learning session, (b) Timebar shall be configured so that time will be adjusted to the available modalities from the learning session, and (c) User can either use the Timebar to jump to specific timestamp or use play/pause functionality and watch a rewind of the entire learning session.



**Figure 3-5 - Timebar Functional Requirements**

Last, but not least, in the Functional category we have Figure 3-6 - Notes Functional requirements, which outline the capability to keep notes, which will be stored and retrieved from a remote server, while studying the data from the modalities in the syncing tool.



**Figure 3-6 - Notes Functional requirements**

Our second table, Table 3-2 - Non-Functional Requirements, lays out the Non-functional Requirements for the Arion System. Like with the functional requirements, we have made categorization, using three categories in this case. Those categories regards Technologies to be used, Performance and Metadata. In detail, the technologies category describes the tech-stack that will be used, namely ReactJS for the front-end, Node.js for the API, MongoDB for the database and the fact that the API shall share a key with the IOLAOS API for authentication. Regarding performance, we have defined 4 requirements, The need for data delivery time between parts of the system to be kept under 1 second, Data processing and response preparation shall be kept under 0.5 seconds, while modalities synchronization time must be kept under one second. Additionally, the system must be able to handle at least 500 simultaneous access requests, so that it will not collapse under the need to serve multiple users at once. Lastly, it is crucial that all modalities are supported by metadata.

**Table 3-2 - Non-Functional Requirements**

Non-functional Requirements		
ID	Title	Description
NT1 <sup>6</sup>	Front-End	The website shall be created with ReactJS
NT2	API	The API shall be created with Node.js
NT3	API key	The API shall share API keys with IOLAOS for authentication
NT4	Database	The Database shall be a MongoDB

<sup>6</sup> nFRs regarding technologies to be used

NP1 <sup>7</sup>	Data Delivery Times	Data delivery time between parts of the system shall be kept under 1 second
NP2	Data Processing	Data processing and response preparation shall be kept under 0.5 seconds.
NP3	Simultaneous Requests	The system shall be able to handle at least 500 simultaneous accesses
NP4	Data Synchronization	Modalities synchronization time shall be kept under 1 second
NSM	Meta-data	All modalities will be supported by meta-data

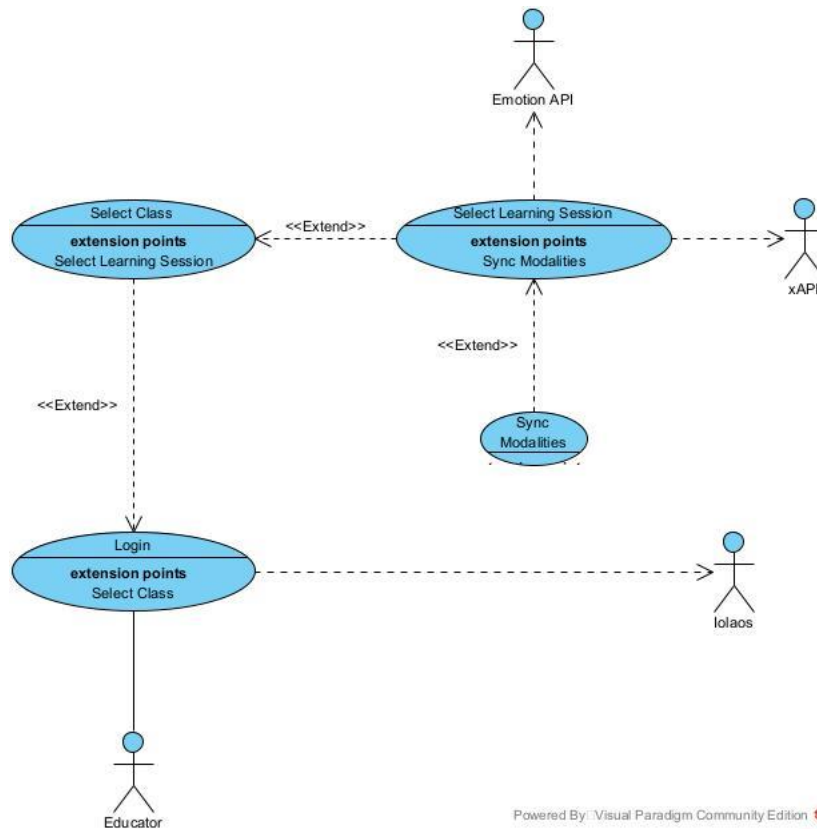
## Use Cases

To better understand the concepts and procedures behind the system we wish to create, and to assist us in creating a more solid implementation plan we studies the uses cases that the system would be subject to, as they emerge from the requirements described above. Here, we will describe the three more prominent use cases, where the end-user (educators) interacts with the system.

Firstly, before any other interaction can be made with the system, the user has to provide his credentials for authentication. This procedure is illustrated in Figure 3-7 - Login to Arion Studio Use Case. After the credentials are provided by the user, IOLAOS API is utilized to authenticate. Assuming the user is valid, a list of classes is retrieved from IOLAOS, the educators select the class she currently wants to work with and is then presented with a list of learning sessions, again selecting the relevant one. After the selection is made, additional data are retrieved from IOLAOS, Emotion API and xAPI, which are loaded to the syncing tool, while the user is forwarded to it.

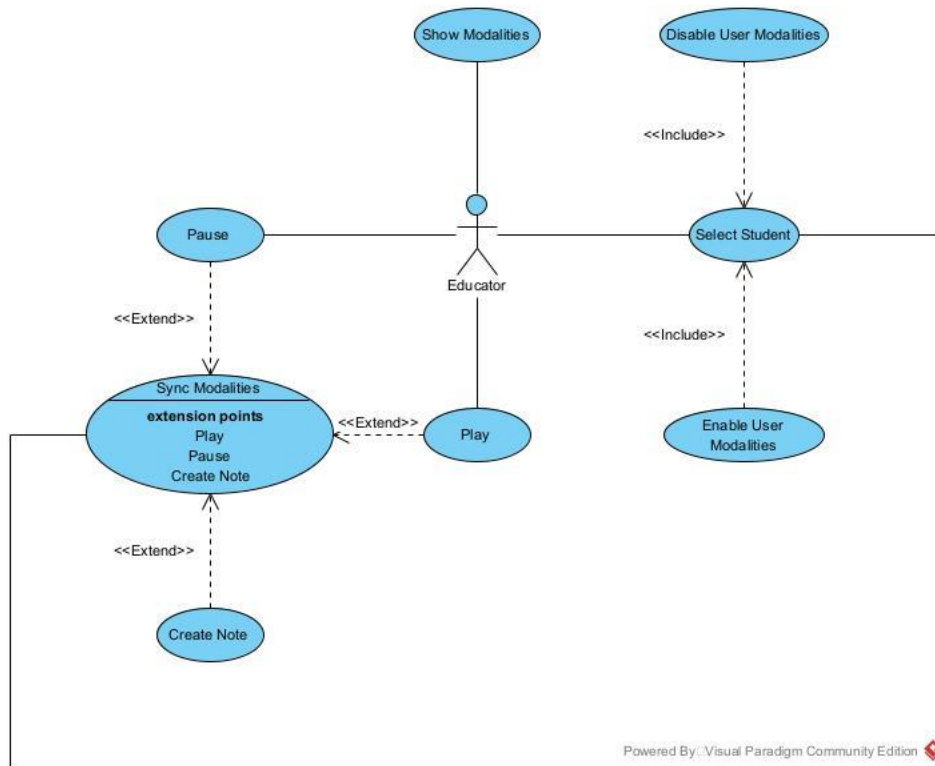
---

<sup>7</sup> nFRs regarding performance



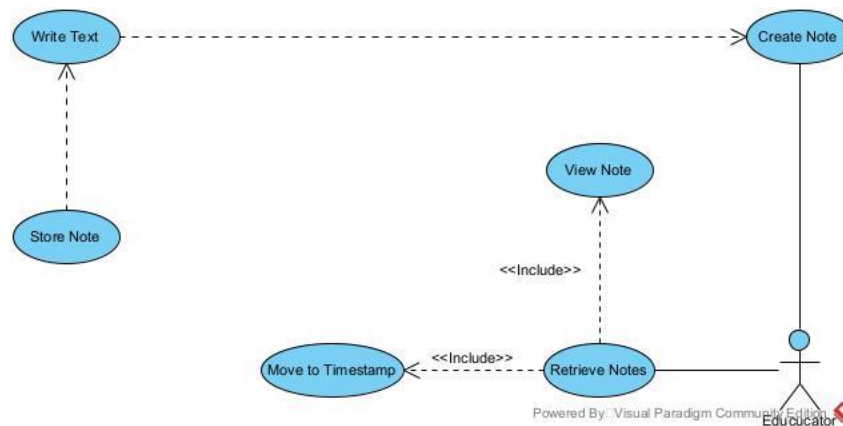
**Figure 3-7 - Login to Arion Studio Use Case**

Second, we have considered the use case of interacting with the Timebar, as it the most significant component of the synchronization tool. Here, the educator, being the user, is the central part of Figure 3-8 - Timebar Use Case. There are a few actions readily available to the educator, she can see the available modalities, selected in the previous use-case, select additional students (or de-select) which will enable an additional action where she can disable or enable modalities to be shown. Additionally, the educator can control the flow of the presented material by playing or pausing, which will in turn trigger the Sync Modalities action, where the modalities are all put into the same timeline, with common start and end points that correspond to the real-world timing of the events. Lastly, we can see the create note action, which is further detailed in the next use case, which is an available option at the Timebar since it is chronically tied to the time reflected by the Timebar.



**Figure 3-8 - Timebar Use Case**

Third, we have the Notes use case, illustrated in Figure 3-9 - Notes Use Case, that follows the actions related to creating, retrieving, and editing notes. In detail, the educator might retrieve notes of the student and learning session she is currently working with, and view (read) the note and move to the timestamp the note is bound to. Additionally, the educator can create a new text, which will lead to a note being created at a particular timestamp, the educator writing the text that she wishes to remember, and the note is then stored at the remote server.

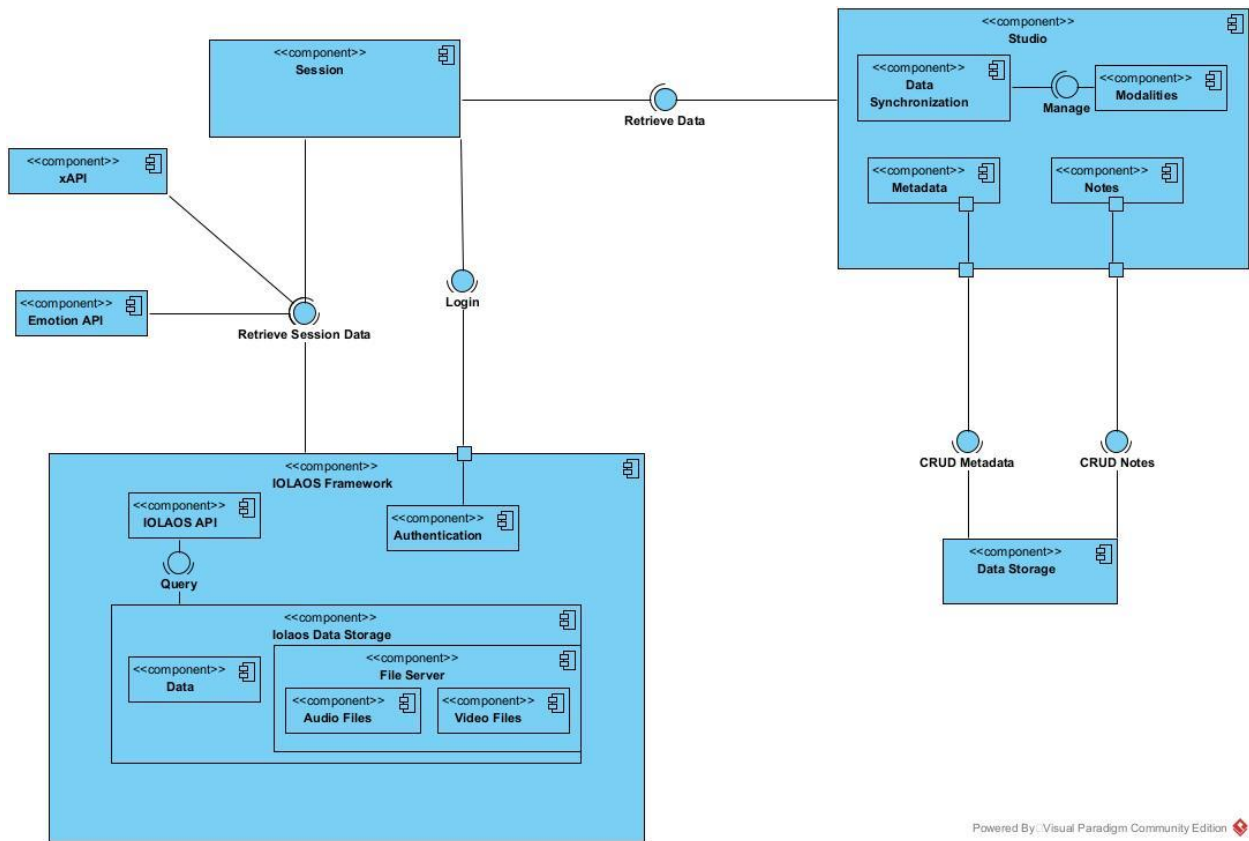


**Figure 3-9 - Notes Use Case**

## **Arion Software Architecture**

After doing the analysis and design described above, we have created an architectural design for the Arion system, from which we then designed the component diagram of Figure 3-10 - Arion Component Diagram. In this we describe the main components of the system and communications with other systems, such as IOLAOS. In the top left corner of the diagram, we have introduced the Studio, the part of the system that users will be interacting with. In this part of the system, we have included the Data Synchronization, which reflect the algorithms and procedures of retrieving data and aligning all data into a common timeline. To achieve this, it manages Modalities, another component of the system. Additionally, through CRUD procedures, the Studio sends and receives data from the Data Storage modality, which is the remote storage of the system. The data that can be saved on the Data Storage reflects two additional modalities, Metadata, which holds information of modalities, and Notes, which holds information and conclusions the educator want to record in a written manner. To make this part of the system work, the Data synchronization needs data to synchronize, thus it retrieves needed data from the Session component, which is the digital representation of learning sessions. Those sessions are introduced by the IOLAOS Framework and is thus a shared component between the Studio and IOLAOS Framework. For each session there are login and retrieval methods, from which we get access to IOLAOS authentication and different needed data, from the common data space or third parties. Through the IOLAOS API we can access data that is stored in the IOLAOS Common Data Space, namely Audio and Video files, user profiles etc. Third parties that will be supported on the first iteration of the Arion System are the Emotion API, which provides data on the emotions of students during learning sessions, extracted from videos stored on the IOLAOS common data space, and the interaction data stored in various LRSs, retrieved through xAPI, that might include virtual classroom data, game data and more.





Powered By: Visual Paradigm Community Edition

Figure 3-10 - Arion Component Diagram

## **Chapter 4 - Implementation**

During this endeavor we implemented the design described above, with a particular focus on data communication between different platforms and the orchestration of scattered data into a single, true to reality, timeline. The need to draw these conclusions led us to the idea of implementing a tool for the synchronization - composition - orchestration of all this valuable untapped data. The aim of this tool is to combine a variety of learning data into simple and understandable forms of information that will lead the teacher to a better understanding of the strengths and weaknesses of students, the lesson, the educational process, and himself by making a critical look at the above with aimed at a substantial improvement of all components of the learning path.

### **Frameworks – Tools & Platforms used**

#### **IOLAOS**

IOLAOS [28], is a versatile, interactive platform that aims to bring various professions closer in the contribution and collaboration for eLearning. IOLAOS is an intricate system with multiple sub-systems, that offer high-quality tools to developers, educators, students, and other professions involved in the learning sphere. An abstract overview of this architecture is illustrated in [28]. Among the most important elements of the platform is the Common Data Center, with is a central repository for data of high educational value. Additionally, IOLAOS offer virtual classrooms, in which educators can meet with their students, using live video, where they can have a virtual lesson, or play educational games. The upside, compared to business-oriented meeting software is that the teacher can have a feed of all students' screens, to monitor their progress and intervene when necessary to help to explain mistakes. Furthermore, IOLAOS has codified learning styles into game preferences which are served through an API, resulting in games that are adjusted with specificity for each individual student. A crucial concept for the IOLAOS system is that of the "Learning Session". This is defined as the single session within a class, where a topic will be examined. If we try to draw a parallel with traditional school, we have the structure of class, meaning the specific students following the same schedule during an entire year, and their teacher, and then we have the learning sessions, meaning a specific meeting with uninterrupted time and specific topic, this could for instance be the history class taken between 10:00 and 10:45 at Monday

the 20<sup>th</sup> of December 2021. Due to the asynchronous nature online learning can have, the time element is not part of the actual session, but we keep the specific topic as the guiding differentiation between sessions. The data saved by IOLAOS, or any collaborating system, to the Common Data Space is saved with the learning session as a primary identifier. Thus, the data saved during learning sessions, including actions in games, video, screen monitoring, chat messages, forum messages, etc., have a continuity in regards of class, topic, learner, educator, and learning sessions. This means that educators can study any angle, with any combination of factors considered. However, achieving this requires skills in statistics, databases, and data visualization that educators most likely do not have. Thus, a need for tools that will make the extraction easy for educators. With all that, IOLAOS is a valuable source of learning session data, like video (face camera and screen record) and video metadata, chat messages, etc. All that data from the learning session are a perfect fit for the Arion synchronization utility. With the synchronization of all that learning session data, Arion works as a complementary utility for the teacher to evaluate the learning process and assign the correct learning preferences to each student.

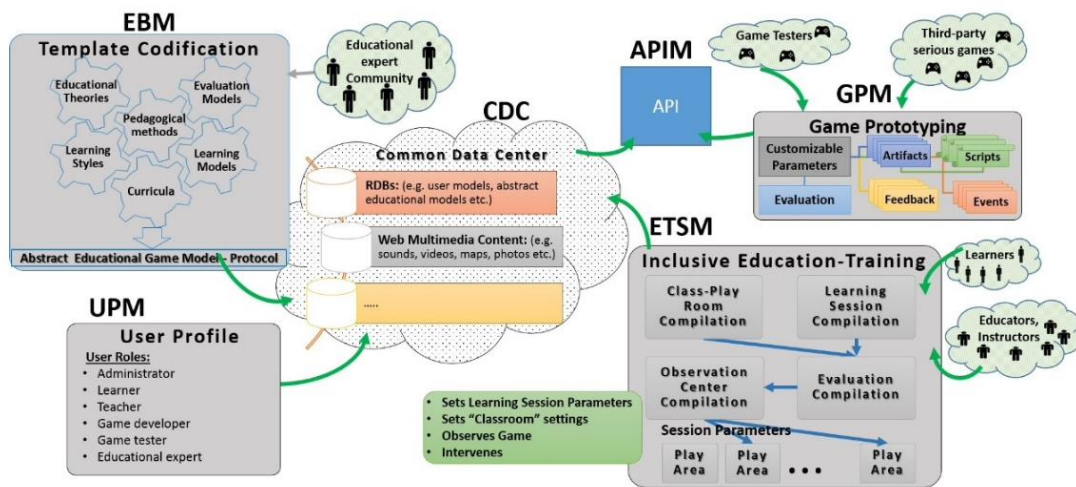


Figure 4-1 - IOLAOS Architecture Overview [28]

## Emotion API

During IOLAOS learning sessions raw data of many kinds are recorded about the procedure. When IOLAOS virtual classroom is used, some additional raw data include face camera

video, screen captures and chat messages. For that reason, the need for analysis and extraction of meaningful information emerged. Emotion API is a tool developed to retrieve this data, analyze it and extract emotional learning analytics, based on the students face expressions, speech and text messages. This is achieved through a chain of events and actions. It is important to note here that the systems provide access on an educator basis, which means that educators can access data only about their classes. If a student participates in classes of two educators, each educator can only retrieve data about this student from her own classes. First within the steps, is to connect with the IOLAOS database, from which classes, learners of the class, personal information and any data recorded during learning sessions will be retrieved. Secondly, xAPI will be used to retrieve data from the LRS. Thirdly, the data retrieved from each request is transformed to the structured required for the best utilization. Fourth step is to feed the transformed data into a machine learning data analysis algorithm, which utilized techniques such as image processing, video analysis, speech sentiment analysis, text sentiment analysis, and more. In this step the algorithm reached a result, which signifies the emotional state per frame, but also evaluates its accuracy. Lastly, the extracted analysis is stored in a database for future retrieval by other collaborating systems, through the systems API. An example of the stored analysis for chat messages can be studied in Table 4-1 - Text sentiment analysis result. Extra data such as id's have been emitted for simplicity and readability. As seen in the table, for each chat message we have the text, this text is given a range, a floating point number, that signifies an emotional range, between -1 and 1, where -1 is the most negative and 1 the most positive, or in other words, happy or sad.

**Table 4-1 - Text sentiment analysis result**

Text	Emotion Range	Emotion Text
Hello!	0	Neutral
My name is Alexis	0	Neutral
I am very happy to play this game with you	0.5095	positive

Thus, we can retrieve students emotional state and orchestrate it with the data feeds from which it has been extracted, as well as any other data available from the particular session. Effectively, to the Arion system this can be yet another input modality, that will further facilitate educators in the evaluation of learning sessions, by complementing existing learning analytics and

acting as a point of reference for emotional responses. Additionally, as a tool that is so closely tied with IOLAOS we feel that it is a necessary modality for our tool, but also a very insightful one that will offer true value to our users.

## **Technologies Used**

### ***MongoDB***

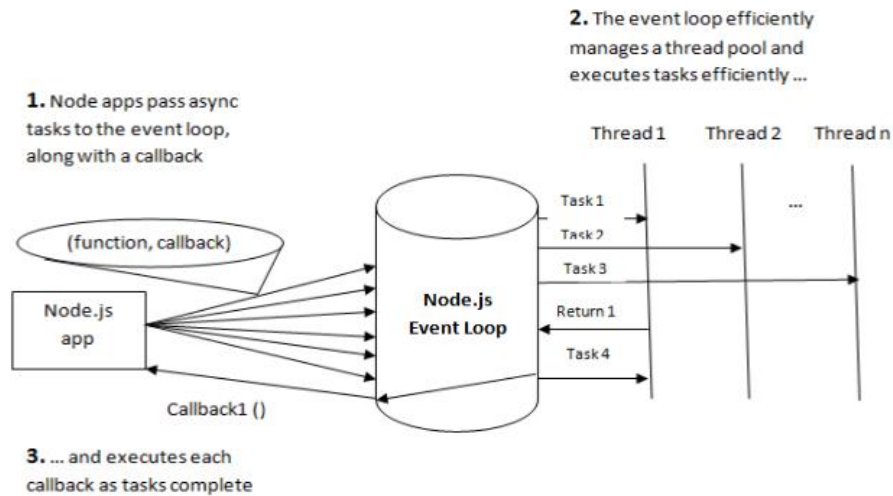
Structured Query Language, or SQL, has been the most extensively used database type in recent decades. The structured data does not allow data duplication and allows relations between different data tables alongside consistent performance makes the SQL languages a beloved tool for Developers. In the last two decades, the amount of data we acquire and the requirement to store unstructured data has grown dramatically. To solve those types of issues introduced first time in 1998 by Carlo Strozzi, the idea of NoSQL meant NO SQL, and after some in 2009 by Eric Evans with this time the meaning has changed to "Not Only SQL" [31]. Key-Value, Document, Column (Big Data databases), and Graph-Oriented databases are the four types of NoSQL databases accessible today[31]. For our database needs we chose MongoDB [32]. It is an open-source database that follows the Not Only SQL approach. As a Document database, MongoDB is scalable and adaptable since the data is stored as JSON documents, which is critical in our scenario because the needs may change quickly. Its strengths lie in the fact that it is an open-source project that follows the non-relational approach to databases. Storage is achieved in the form of JSON documents, which makes any MongoDB database very scalable and flexible, which is quite important to our case since the needs might quickly change. Additionally, due to the JSON format used, there are no data transformations needed between parts of our system, as the back-end and front-end will be developed with JavaScript frameworks. This saves both in computational resources and in time spent designing and coding. Of course, being a schema-less database comes with drawbacks, not only benefits. There is no way to ensure consistency in data, which means that developers using said data must be careful and have proper mechanisms to handle possible errors and exceptions. Additionally, if the database grows a lot, performance can quickly become an issue [33]. However, MongoDB is generally accepted as a better alternative when a flexible data model is crucial to store denormalized data [34]. Furthermore, because our systems (Frontend and Backend) are developed with JavaScript frameworks, there will be no data transformations

required between components of our system due to the JSON format. Except for all these positive aspects, MongoDB has some flaws. Because there is no way to guarantee data consistency, developers who use it must be cautious and have appropriate methods in place to manage potential errors and exceptions [35]. However, MongoDB is generally accepted as a better alternative when a flexible data model is crucial to store denormalized data [31]. In the case of Arion, we believe the benefits to be crucial to the implementation, and the disadvantages are unimportant to us because we seek to build a microservices architecture that will never allow the database to expand out of proportion.

### *NodeJS*

JavaScript is a well-known programming language that has familiarized its name with the web-development alongside HTML, CSS. Back in the days of the first browsers, JavaScript made its appearance with the browser Netscape Navigator. Except for the frontend use of JavaScript, which is generally known, the Netspace server-side business model utilized JavaScript in server-side. Unfortunately, JavaScript on the server-side failed to succeed until the recent introduction of Nodejs. Nodejs was born back in 2009, and today, it is adopted by some of the world's largest corporations, including LinkedIn and Uber [36], [37]. NodeJS [38] is a runtime environment for JavaScript, often used for server-side programming. It is built with an open source WebAssembly engine, called V8, that is published by google [39] as part of the Chromium project. It has become very popular in recent years as it is a unique way to utilize JavaScript for server-side programming, which is a breakthrough for a language that has so far been used for client-side programming, handling UI elements and dynamic web pages. There has been critiques though, as the result is a single-threaded server, while most other languages follow a multi-threaded paradigm, creating servers that are able to handle parallel requests with better optimization and carry computationally heavy procedures. Nodejs does not use multithreading to avoid complex and spaghetti code and avoid problems like isolation failures, deadlocks and more. The runtime environment of Nodejs works with an asynchronous I/O event model. This model works as single-threaded like JavaScript work, to process all incoming events-requests does not use multithreading but uses a non-blocking event loop. This non-blocking event loop accepts virtually all the incoming requests and handles them with an intelligent algorithm built with an asynchronous architecture. This architecture uses callbacks to handle all requests and prevents clock cycles from being wasted. Specifically, it uses

a non-blocking event-driven loop that handles virtually in parallel all the available events to be able to handle them efficiently with the JavaScript single-threaded paradigm [37] as we see in Figure 4-2 Node.js Architecture.



**Figure 4-2 Node.js Architecture**

The benefits seem to outweigh those weak points of the language though, since the Node Package Manager [40] is allowing programmers to quickly and effortlessly import node modules into their applications, thus making it very easy to reuse code and reach results faster. Additionally, utilizing the same language for the server and client-side programming is very attractive to developers that can now focus more on one language. It is not suitable for large scale applications, with millions of request or CPU intensive subroutines [33], but it is very commonly preferred for applications in the microservices architecture, where instead of creating one very large application to handle everything, the system is instead distributed to as small services as possible, and other small-scale projects. In the case of Arion, we consider NodeJS to be a very suitable technology, as we wish to follow the microservices logic, and much of the server-side programming is handling connections to other APIs, while additionally NodeJS will allow us to create a fast and reliable RESTful API that will control all of the platform's data flow and assets.

## ***ReactJS***

ReactJS [41] is a powerful open-source JavaScript library that helps developers build Single-Page-Applications (SPAs) with dynamic user interfaces. React is typically used to create large web applications using JavaScript. In general, it is a component-based library that seeks to construct web applications composed of minimal components using the Single Responsibility Technique [42]. React provides a broad number of features that help you develop powerful web apps [43]. Some features are: (a) JSX: React, in contrast with a simple web application written in HTML, CSS, and JS does not use HTML elements to create the user interface elements but utilizes JavaScript language syntax is JSX (JavaScript Syntax Extension) and referred to as JavaScript XML.JSX is a popular language that allows developers to effortlessly develop mark-up components and handle all the events using JavaScript . (b) Easy Learn & Learn Once, Write Anywhere: One important feature is the easy learning curve. The use of the JSX language, the non-strict structure, and the easily adaptable architecture without making assumptions about the rest of the technology stack makes React a library with an easy learning curve. (c) Props, State & lifecycles hooks: React manages all the events and the DOM manipulation with the help of Props and State. To begin with, data flow in a React app, Props are properties (values or handlers) that can be handed down from a parent component to its children. Additionally, react utilizes the State, which is a JavaScript object that can influence component behavior, to handle all the changes and reflect them to the DOM. To handle the component behavior also introduces lifecycles hooks, which are methods that are triggered in events like component mount and unmount and allow to write the logic for those specific events. (d) Lightweight Virtual DOM: The Document Object Model (DOM) is a programming interface that browsers use to represent a web page as a document with structure and style. Unlike React, many frameworks that do not intend to create single-page web applications communicate directly with the DOM. This direct approach results in constant whole DOM manipulation in every trigger event, and a large amount of data needs to be loaded again. To avoid this substantial performance impact, React avoids that approach. React uses a DOM similar to the browser DOM but stored in memory, called Virtual DOM. The React library, every time the state changes or a request to change the view of the web page, compares the Virtual DOM with the Browser DOM. When a difference is discovered, React manipulates the browser DOM so the visual difference can be visible. These comparisons are made using the diff algorithm, and the results are reflected in the DOM tree. (e) React is designed with a component-based



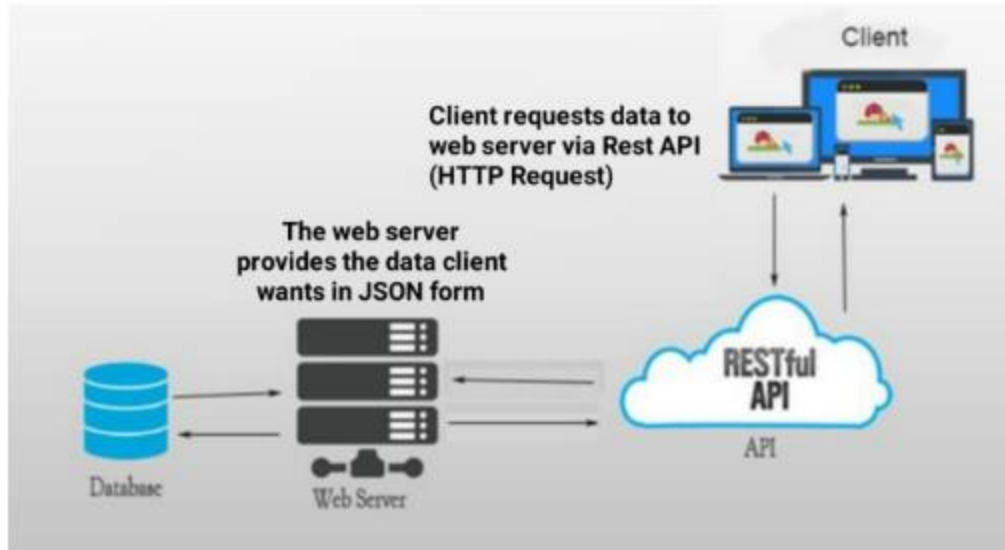
hierarchy architecture. With that type of architecture, every component inherits its data from the parent. Having all that in mind, we understand that ReactJS cannot have a unidirectional data flow because then the data cannot be immutable. React's one-way data flow and Flux architecture (Flux is an alternative for the standard model view controller (MVC)) enable the ability to trigger User Interface modifications in response to data changes. (f) Redux: With not having two-way data binding and the need to have data immutable, React except Props uses Redux. Redux is a "predictable state container"[4]. It functions as a "single source of truth" that allows all components to access the State.[4] The Redux consists of:

1. Store: The Store is a global State container holding that has a get function that retrieves the current State. .[4]
2. Actions: The Actions are JavaScript objects that contain the State that needs to be stored and usually generated by an action creator (UI button).[4]
3. Reducers: Reducers are methods that get an Action as a parameter, and they transform the State and return the newly modified State.[4]
4. Middlewares: Middlewares are methods for executing logic that must be performed between actions and reducers.[4]

All these features make React a tool with high performance. This characterization comes from the high efficiency coming from the non-direct browser DOM manipulation. With the use of the Virtual DOM in memory and the comparison of two with the help of diff algorithm that allows manipulating the browser DOM when is mandatory (Only the nodes of the browser's DOM tree that are relevant and desired are updated).

### ***REST API & Databases***

As described in the previous sections, our server-side application implementation is developed with the help of Nodejs, and our data are collected in a NoSQL database called MongoDB. With the way we designed our backend, a Restful API is the best fit for our use case. Restful API uses the REST (Representational State Transfer) architecture, illustrated in Figure 4-3 - Generic RESTful API Architecture, which uses the HTTP (Hypertext Transfer Protocol) layer to develop web services, acting as a communication layer between the client and the database or other services.



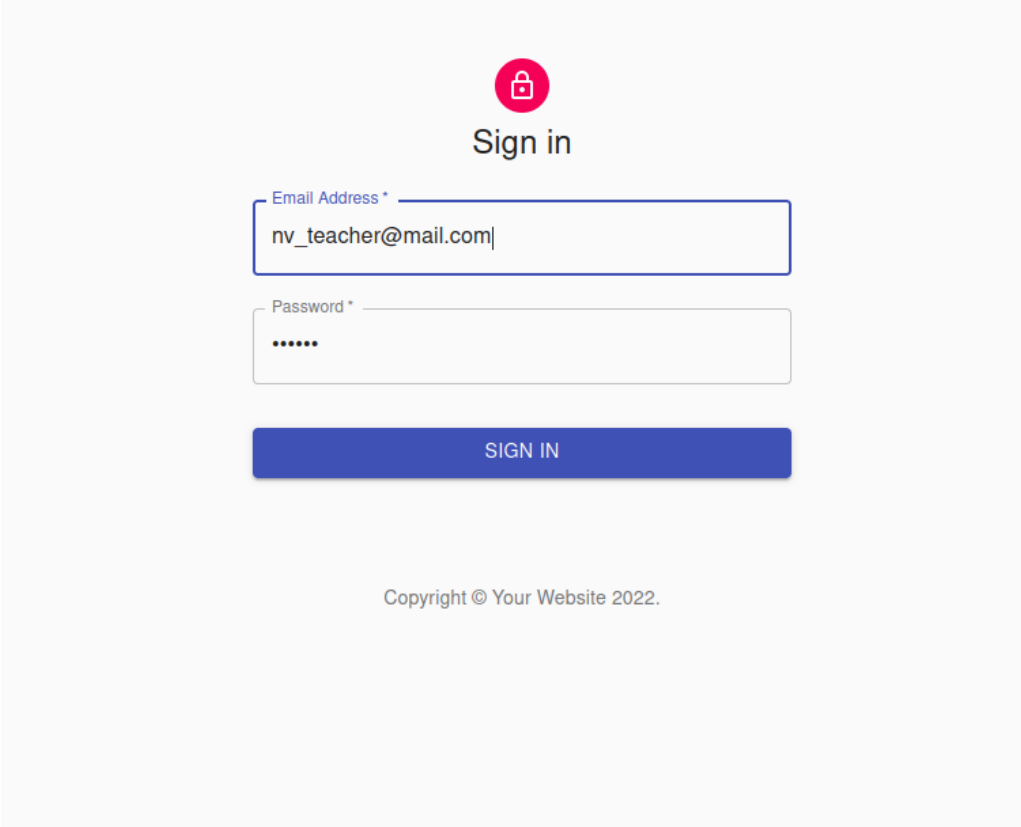
**Figure 4-3 - Generic RESTful API Architecture**

Arion platform Restful API's goal is to retrieve data like video, chat, sound, xAPI, or even emotion processed data, process them, and provide them to the client for synchronization. Thus, operating as an agnostic API that gets data by communicating with IOLAOS API. After getting data from the external API of IOLAOS, process the video to extract metadata, helpful for the synchronization process. After collecting and processing data from IOLAOS API continues to retrieve data from API about the games played in the IOLAOS learning sessions and the data from the processed video of the learning session by the Emotion API to provide them to the client for synchronization. Aside from the retrieval process, stores notes and changes generated by educational experts' interactions with the client timeline. Admittedly, the API is a small implementation but is designed to be easily scalable and maintainable for further development.

## Chapter 6 - Pilot Use Case

The Pilot use case chapter is a very remarkable chapter that showcases a descriptive scenario of the Arion platform. With this scenario, we can thoroughly illustrate all the aspects, the benefits, and the reasons for someone to use the Arion platform.

Our scenario starts with a teacher who schedule's a Learning session (an online version of a classroom lesson) on the IOLAOS platform for the GameHub class. After the Learning Session is concluded, the Learning Analytics material that has been generated by the IOLAOS platform, like screen or camera video and chat messages are stored in the IOLAOS common data space. The teacher's goal is to assess his/her decisions regarding the learning preferences assigned to the students who took part in the Learning session. To achieve this goal, the teacher joins the Arion platform to evaluate the learning process through the synchronization utility that the platform provides, utilizing the learning analytics materials stored by IOLAOS platform during the learning session.

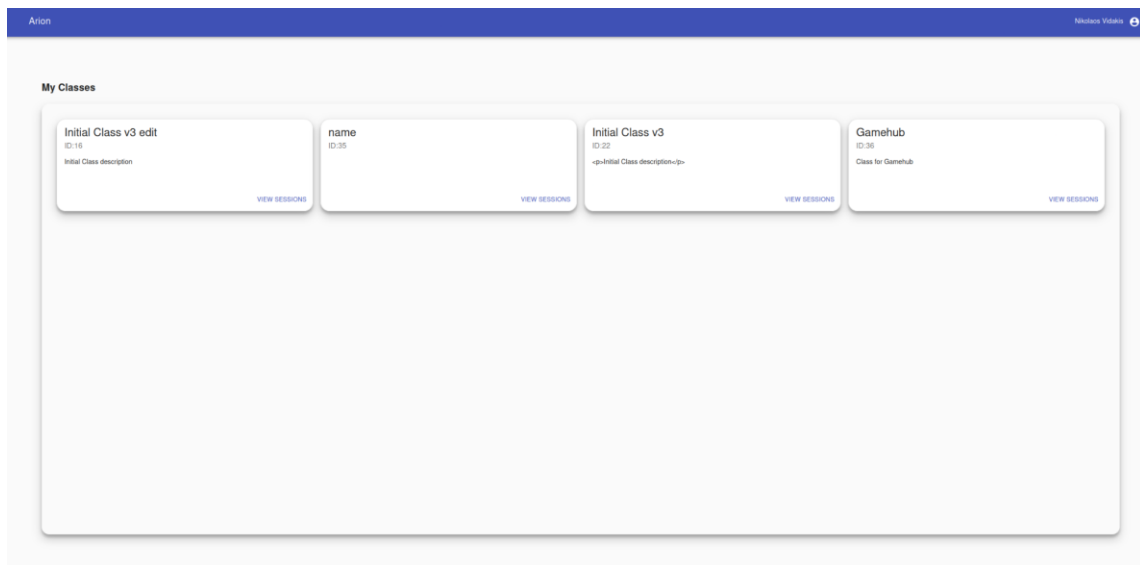


The image shows a login form for the Arion Platform. At the top center is a red circular icon with a white padlock. Below the icon is the text "Sign in". There are two input fields: the first is labeled "Email Address \*" and contains the text "nv\_teacher@mail.com"; the second is labeled "Password \*" and contains six dots. Below the input fields is a blue button with the text "SIGN IN". At the bottom center of the form is the text "Copyright © Your Website 2022."

**Figure 6-1 - Arion Platform login form**

As the first step after wrapping up the learning session in the IOLAOS platform, the teacher starts the evaluation process through the Arion platform. The teacher goes to the Arion platform synchronization utility's homepage and fills out the login form (shown in Figure 6-1 - Arion Platform login form). The teacher fills out the credentials needed to access the IOLAOS platform in that form and authentication goes through the IOLAOS platform authentication system, with the use of services. As a result, at this stage of the implementation, every teacher (Arion user) must be registered on the IOLAOS platform.

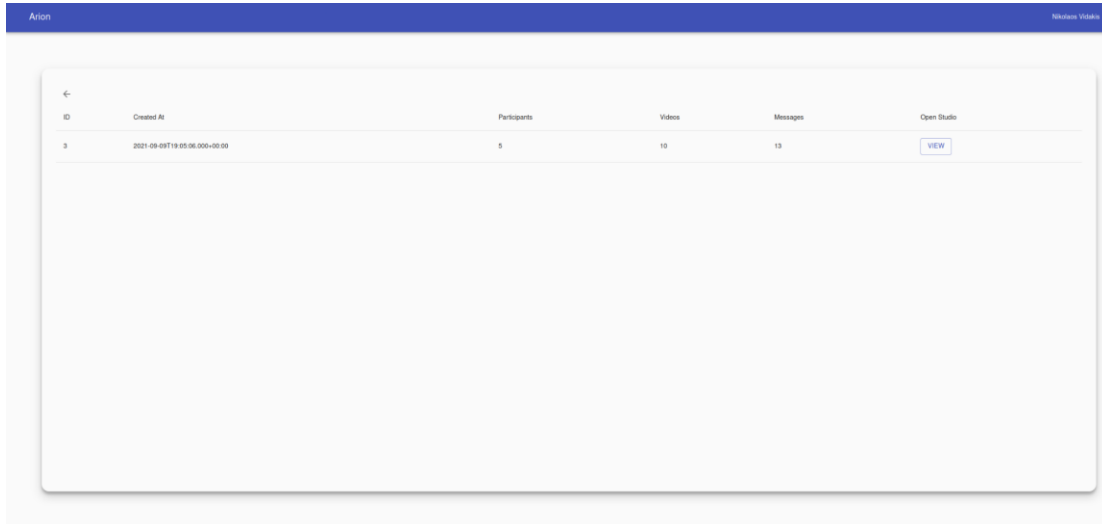
After a successful login in the platform, all the classes the teacher is responsible for or is an assistant for are displayed. With that view, the teacher has an organized view of all the classes and can easily navigate to the correct class for which he/she aims to evaluate one or more learning sessions. The classes are displayed as cards in a grid view, as shown in Figure 6-2 - Available Classes View, and they can choose through the card button to see the available learning session.



**Figure 6-2 - Available Classes View**

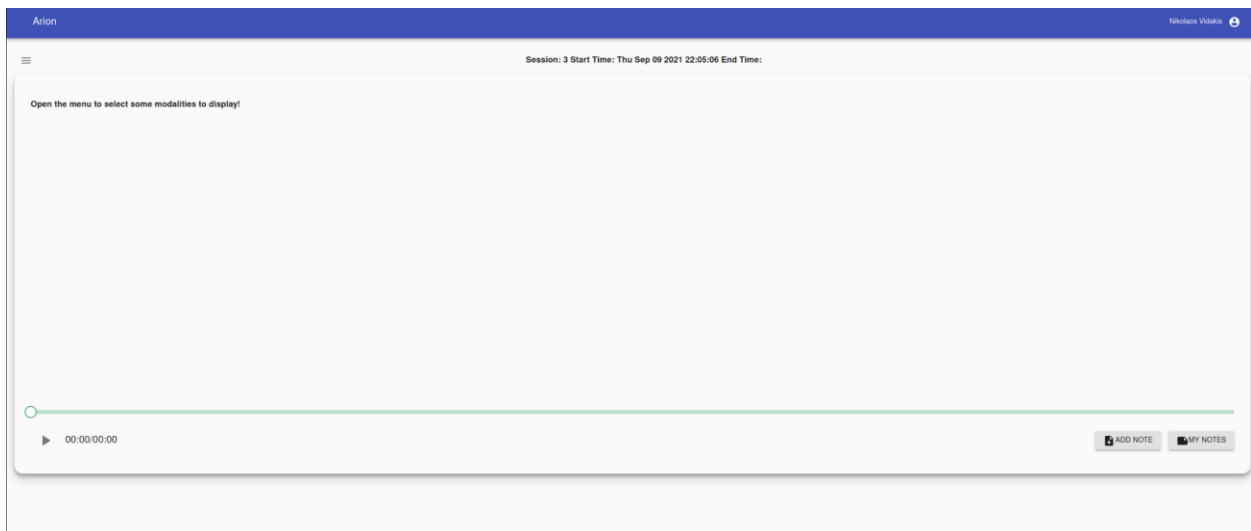
Furthermore, after choosing the GameHub class from the available classes view (Figure 6-2 - Available Classes View), the teacher can see all the available learning sessions with some basic information about the creation date, the participants, the available videos, and chat messages like Figure 6-3 - Available Class Learning sessions shows. After choosing our scenario's latest learning

session, the teacher presses the view button to load the synchronization view with the available data from the learning session.






**Figure 6-3 - Available Class Learning sessions**

Initially the synchronization view (Open studio) displays an empty player, similar to those found on video platforms such as YouTube, with two buttons that allow the teacher to create notes and view the notes he has already created. Additionally, as we see in Figure 6-4 - Open Studio View, the top left corner appears to have a "burger" icon through which a menu of necessary tools are accessed.



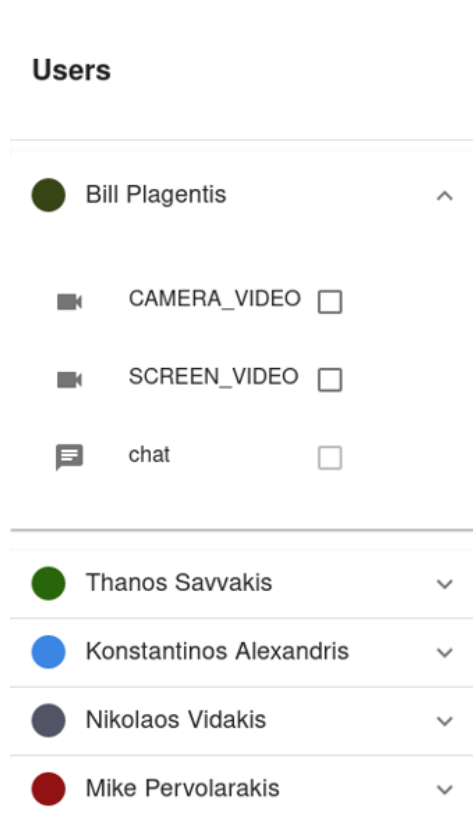
**Figure 6-4 - Open Studio View**

That view is a canvas for all the modalities of the platform. The platform supports various modalities except for the video modality that supports screen and camera video. The studio view has three modalities (shown in Figure 6-5 - Arion Platform Modalities), the first of which is the chat modality, which allows the platform to display the chat for all or specific participants. The second modality is the xAPI, which is a collection of statistics derived from the games that participants played throughout the learning session. Finally, the emotions modality, which is derived from the emotion API, shows the results of the emotion analysis of all videos and chats from the learning session.

Modalities		
	Chat	<input type="checkbox"/>
	xAPI	<input type="checkbox"/>
	Emotions	<input type="checkbox"/>

**Figure 6-5 - Arion Platform Modalities**

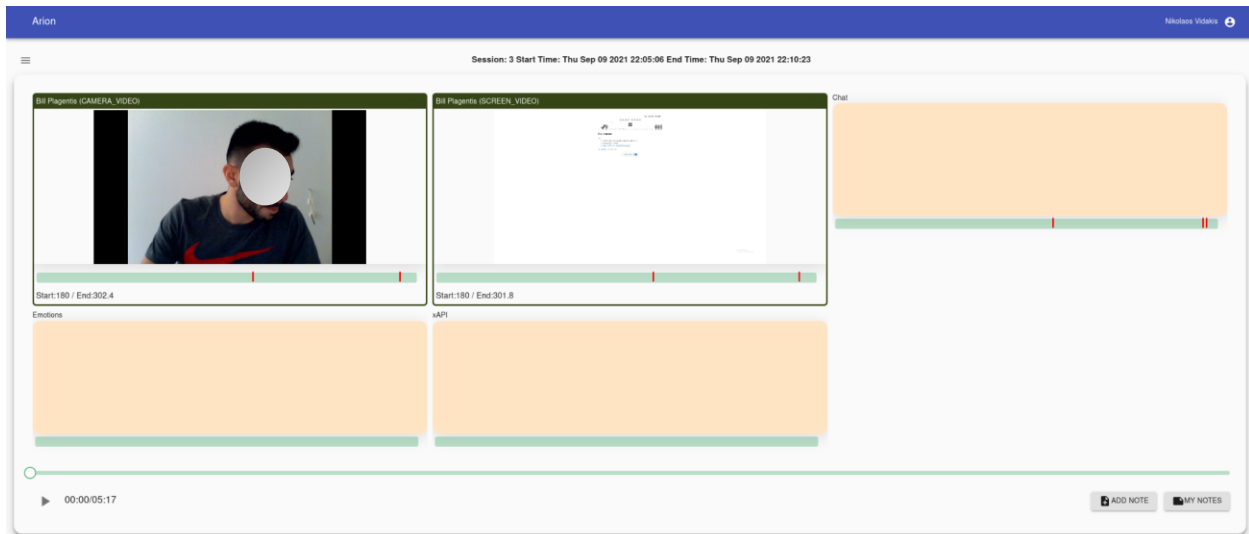
To access all those modalities, the user can click the burger icon from the top left area of the UI interface Figure 6-4 - Open Studio View. The user can enable the general modalities from the bottom area, as shown in Figure 6-5 - Arion Platform Modalities, and the user-specific modalities from the top section, as shown in Figure 6-6 - User Specific Modalities. The top section provides all the available users, and for each one of them the teacher has the ability to enable the user-specific chat (needs to be first enabled from the bottom section), the camera, and the screen video modalities.



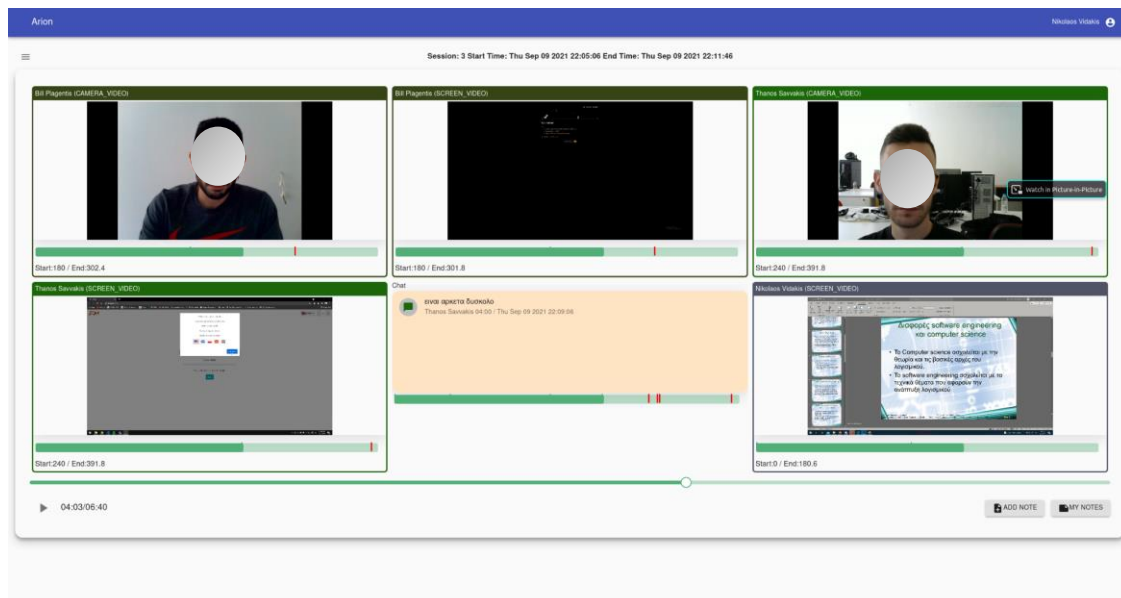
**Figure 6-6 - User Specific Modalities**

After selecting the platform's available modalities, the UI places them in the main part of the open studio and arranges them in a grid-like view where each modality is a tile, as shown in Figure 6-7 - Open Studio with all the possible modalities.

The grid view at the current state supports a dynamic grid with a maximum size of 2x3. The grid also has dynamic features like drag and drop that allow changing the order of the available modalities. All of this enables the teacher to create a specific order of modalities, allowing the teacher, for example, to have the videos of the users' camera or screen in the order that helps to evaluate the learning process. After completing the process of adding modalities and ordering them, the teacher has a synchronized timeline based on the first and the last event from the enabled modalities, as we see in Figure 6-8 - Open Studio with ordered modalities.



**Figure 6-7 - Open Studio with all the possible modalities**

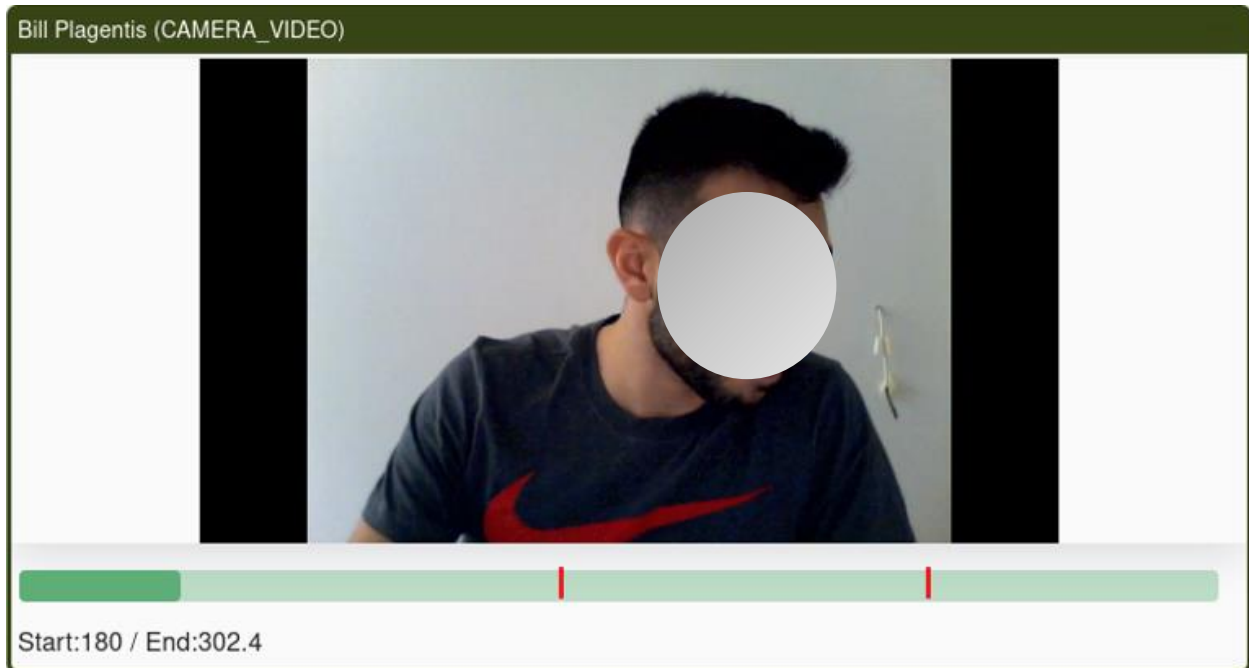


**Figure 6-8 - Open Studio with ordered modalities**

Furthermore, the teacher can start the timeline from the player of Open Studio and move the timeline to a specific time. For example, if the teacher wants to move the time at the start of the video, the teacher moves the timeline button. Every modality has a red indicator in the sub timeline



of the modality tile to help the user spot the available events and locate the video starting point, as shown in Figure 6-9 - Camera Video modality with start and end event.



**Figure 6-9 - Camera Video modality with start and end event**

In addition to the timeline setup and the navigation, the teacher can note the timeline at specific times that he/she considers to be important. A red line appears in the timeline while a note is being created, indicating the note's appearance. That indicator gives the teacher the ability to recognize the frame in which the note was created, and it aids the teacher in remembering the remarkable events. The note creation is able through a button in the bottom right area of the UI (Figure 6-7 - Open Studio with all the possible modalities) that opens a modal, as Figure 6-10 - Note Creation Modal shows, and allows the teacher to type a message and save the note.

Finally, the teacher can access all the created notes and edit them through the My Notes bottom right area of the UI (Figure 6-7 - Open Studio with all the possible modalities), which opens a modal (Figure 6-11 - Available Notes Modal) that shows the available notes and provides the edit and delete functions.



## Create Note

Add Note



CREATE NOTE AT 05:07

Figure 6-10 - Note Creation Modal



## My Notes

02:35  

4/10 φοιτητές δεν παρακολουθούν το μάθημα. Παρατηρείτε μεγάλη κινητικότητα στο διαδίκτυο

04:51  

Απο τα συναισθήματα των φοιτητών συμπαιρένουμε ότι το μάθημα είναι αποδοτικό

05:07  

Η παράδοση θέλει βελτίωση στα slide απο 5-8 διότι παρατηρείτε μη διατήρηση προσοχής των φοιτητών

Figure 6-11 - Available Notes Modal

## **Chapter 7 - Conclusion & Future Work**

Education is an evolving area, and increasing numbers of researchers are studying new pedagogical methods, learning styles, etc. Many new technologies, such as serious games, MOOCs, and technologies for game-based learning, are booming alongside the researchers. Based on our previous experience with educational technologies such as the IOLAOS platform, we've discovered an area with plenty of space for new technologies to aid in the education process. That area is more focused on learning process evaluation is a very important process in education because it aids in the prevention of issues in the learning process as well as the appearance of improvements. With that idea in mind, we designed and implemented a platform that aims to synchronize media like camera and screen video alongside other data like chat, xAPI data from game-based learning, and emotion data from the analyzed videos. With all that material now synchronized, we have a fantastic tool that allows teachers that use the IOLAOS platform to assess and improve the learning process.

In the future, we aim to improve the Arion platform in all its aspects. First and foremost, we want to expand the supported media files. At the moment, the platform only supports video. The next step is to add audio support and the ability to analyze audio, similar to how other platforms analyze the sound to generate emotion based on it. Furthermore, the platform supports video only through the IOLAOS platform, videos generated from the learning session, and can be camera or screen videos. Having only that way of importing media into the platform makes the platform only accessible for a small number of users, therefore we aim to make it accessible for a broad range of users and broadening the import options. To achieve this, we aim to support third-party platforms such as YouTube and the ability to import user videos. Finally, we want to improve the UI usability, and specifically, the modalities grid that is currently supported only supports a maximum of 2x3 grid modalities to be dynamic.

## References Or Bibliography

- [1] C. C. Abt, "Serious Games." 1970. doi: 10.1109/VS-GAMES.2009.8.
- [2] M. Ulicsak, "Games in Education: Serious Games," *A FutureLab Literature Review*, p. 139, 2010.
- [3] M. Zyda, "From visual simulation to virtual reality to games," *Computer*, vol. 38, no. 9, pp. 25–32, 2005, doi: 10.1109/MC.2005.297.
- [4] S. Arnab *et al.*, "Mapping learning and game mechanics for serious games analysis," *British Journal of Educational Technology*, vol. 46, no. 2, pp. 391–411, 2015, doi: 10.1111/bjet.12113.
- [5] F. Bellotti, B. Kapralos, K. Lee, P. Moreno-Ger, and R. Berta, "Assessment in and of serious games: An overview," *Advances in Human-Computer Interaction*, vol. 2013, 2013, doi: 10.1155/2013/136864.
- [6] M. Paulsen, "Online Education Systems : Discussion and Definition of Terms," *NKI Distance Education*, pp. 1–8, 2002.
- [7] C. Hurtado, G. Licea, and M. Garcia-Valdez, "Integrating Learning Styles in an Adaptive Hypermedia System with Adaptive Resources," *Studies in Systems, Decision and Control*, vol. 143, pp. 49–67, 2018, doi: 10.1007/978-3-319-74060-7\_3.
- [8] P. Brusilovsky, "Adaptive hypermedia for education and training," *Adaptive Technologies for Training and Education*, pp. 46–66, 2012, doi: 10.1017/CBO9781139049580.006.
- [9] X. Liu, Y. Liu, H. Song, and A. Liu, "Big Data Orchestration as a Service Network," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 94–101, 2017, doi: 10.1109/MCOM.2017.1700090.
- [10] C. Peltz, "Web services orchestration and choreography," *Computer*, vol. 36, no. 10, pp. 46–52, Oct. 2003, doi: 10.1109/MC.2003.1236471.
- [11] I. Tabak, "Synergy: A Complement to Emerging Patterns of Distributed Scaffolding," *Journal of the Learning Sciences*, vol. 13, no. 3, pp. 305–335, Jul. 2004, doi: 10.1207/s15327809jls1303\_3.
- [12] S. Ryan, "What is data orchestration," Dec. 15, 2020. <https://www.mparticle.com/blog/data-orchestration> (accessed Jan. 26, 2022).

- [13] G. Natriello, “Imagining, seeking, inventing: the future of learning and the emerging discovery networks,” *Learning Inquiry*, vol. 1, no. 1, pp. 7–18, May 2007, doi: 10.1007/s11519-007-0005-1.
- [14] M. Watts, “The orchestration of learning and teaching methods in science education,” *Canadian Journal of Science, Mathematics and Technology Education*, vol. 3, no. 4, pp. 451–464, Oct. 2003, doi: 10.1080/14926150309556582.
- [15] P. Dillenbourg and P. Jermann, “Designing Integrative Scripts,” in *Scripting Computer-Supported Collaborative Learning*, Boston, MA: Springer US, 2007, pp. 275–301. doi: 10.1007/978-0-387-36949-5\_16.
- [16] J. Liang, P. Huang, J. Chen, and A. Hauptmann, “Synchronization for multi-perspective videos in the wild,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, pp. 1592–1596. doi: 10.1109/ICASSP.2017.7952425.
- [17] J. Kammerl *et al.*, “Temporal synchronization of multiple audio signals,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 4603–4607. doi: 10.1109/ICASSP.2014.6854474.
- [18] C. Lu and M. Mandal, “An efficient technique for motion-based view-variant video sequences synchronization,” in *2011 IEEE International Conference on Multimedia and Expo*, Jul. 2011, pp. 1–6. doi: 10.1109/ICME.2011.6011973.
- [19] P. Shrestha, H. Weda, M. Barbieri, and D. Sekulovski, “Synchronization of multiple video recordings based on still camera flashes,” in *Proceedings of the 14th annual ACM international conference on Multimedia - MULTIMEDIA '06*, 2006, p. 137. doi: 10.1145/1180639.1180679.
- [20] P. Shrstha, M. Barbieri, and H. Weda, “Synchronization of multi-camera video recordings based on audio,” in *Proceedings of the 15th international conference on Multimedia - MULTIMEDIA '07*, 2007, p. 545. doi: 10.1145/1291233.1291367.
- [21] J. Liang, S. Burger, and A. Hauptmann, “Video Synchronization and Sound Search for Human Rights Documentation and Conflict Monitoring,” 2018.
- [22] K. Su, M. Naaman, A. Gurjar, M. Patel, and D. P. W. Ellis, “Making a scene,” in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval - ICMR '12*, 2012, p. 1. doi: 10.1145/2324796.2324829.

- [23] M. Aparicio and C. J. Costa, “Data visualization,” *Communication Design Quarterly*, vol. 3, no. 1, pp. 7–11, Jan. 2015, doi: 10.1145/2721882.2721883.
- [24] M. Friendly, “A Brief History of Data Visualization,” in *Handbook of Data Visualization*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 15–56. doi: 10.1007/978-3-540-33037-0\_2.
- [25] L. Po, N. Bikakis, F. Desimoni, and G. Papastefanatos, “Linked Data Visualization: Techniques, Tools, and Big Data,” *Synthesis Lectures on the Semantic Web: Theory and Technology*, vol. 10, no. 1, pp. 1–157, Mar. 2020, doi: 10.2200/S00967ED1V01Y201911WBE019.
- [26] Z. Wang, L. Sundin, D. Murray-Rust, and B. Bach, “Cheat Sheets for Data Visualization Techniques,” in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Apr. 2020, pp. 1–13. doi: 10.1145/3313831.3376271.
- [27] T. Anya and E. Lias, “Learning Analytics: Definitions, Processes and Potential,” 2011.
- [28] N. Vidakis and S. Charitakis, “Designing the Learning Process,” in *Proceedings of the 10th International Conference on Subject-Oriented Business Process Management - S-BPM One '18*, 2018, pp. 1–11. doi: 10.1145/3178248.3178254.
- [29] Rustici Software, “Experience API.” <https://xapi.com/overview/> (accessed Jan. 26, 2018).
- [30] A. M. Trampas, “EXTRACTING LEARNING ANALYTICS FROM GAME BASED LEARNING SESSIONS GENERATED BY MULTIMODAL DATA SOURCES,” 2021.
- [31] C. Gyorodi, R. Gyorodi, G. Pecherle, and A. Olah, “A comparative study: MongoDB vs. MySQL,” in *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*, Jun. 2015, pp. 1–6. doi: 10.1109/EMES.2015.7158433.
- [32] MongoDB, “MongoDB.” <https://www.mongodb.com/> (accessed Apr. 13, 2021).
- [33] M. Satheesh, B. J. Dmellon, and J. Kron, *Web Development with MongoDB and NodeJS - Second Edition: Build an interactive and full-featured web application from scratch using Node.js and MongoDB*, Second. Birmingham: Packt Publishing Ltd, 2015.
- [34] C. Gyorodi, R. Gyorodi, G. Pecherle, and A. Olah, “A comparative study: MongoDB vs. MySQL,” in *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*, Jun. 2015, pp. 1–6. doi: 10.1109/EMES.2015.7158433.
- [35] M. Satheesh, B. J. D’mello, and J. Krol, *Web Development with MongoDB and NodeJS*, 2nd Edition. Packt Publishing, 2015.

- [36] “A brief History of Node.js.” <https://nodejs.dev/learn/a-brief-history-of-nodejs> (accessed Jan. 16, 2022).
- [37] S. Tilkov and S. Vinoski, “Node.js: Using JavaScript to Build High-Performance Network Programs,” *IEEE Internet Computing*, vol. 14, no. 6, pp. 80–83, Nov. 2010, doi: 10.1109/MIC.2010.145.
- [38] “NodeJS.” <https://nodejs.org/> (accessed Apr. 13, 2021).
- [39] Google, “V8 Engine.”
- [40] “About NPM.” <https://docs.npmjs.com/about-npm> (accessed Jan. 16, 2022).
- [41] “ReactJS.” <https://reactjs.org/> (accessed Jul. 15, 2021).
- [42] “Thinking in React.” <https://reactjs.org/docs/thinking-in-react.html> (accessed Jan. 16, 2022).
- [43] S. Fitzgerald, *React js: Modern web development using React js*. 2020.