

Article

# Moodle LMS Integration with Amazon Alexa: A Practical Experience

Javier Ochoa-Orihuel <sup>1,\*</sup>, Raúl Marticorena-Sánchez <sup>1</sup> and María Consuelo Sáiz-Manzanares <sup>2</sup>

<sup>1</sup> Escuela Politécnica Superior, Universidad de Burgos, Campus Vena, Avda. Cantabria, s/n, 09006 Burgos, Spain; rmartico@ubu.es

<sup>2</sup> Facultad de Ciencias de la Salud, Universidad de Burgos, P<sup>o</sup> Comendadores, s/n (Hospital Militar), 09001 Burgos, Spain; mcsmanzanares@ubu.es

\* Correspondence: joo0003@alu.ubu.es

Received: 8 September 2020; Accepted: 26 September 2020; Published: 29 September 2020



**Abstract:** The frequency of interaction between teachers and students through Learning Management Systems (LMSs) is continuously rising. However, recent studies highlight the challenges presented in current LMSs to meet the specific needs of the student, regarding usability and learnability. With the motivation to support the research of effectiveness when using a Voice User Interface (VUI) for education, this paper presents the work done (RQ1) to build the basic architecture for an Alexa skill for educational purposes, including its integration with Moodle, and (RQ2) to establish whether Moodle currently provides the necessary tools for voice-content creation for develop voice-first applications, aiming to provide new scientific insight to help researchers on future works of similar characteristics. As a result of this work, we provide guidelines for the architecture of an Alexa skill application integrated with Moodle through safe protocols, such as Alexa's Account Linking Web Service, while our findings ratify the need for additional tooling within Moodle platform for voice-content creation in order to create an appealing voice experience, with the capabilities to process Moodle data structures and produce sensible sentences that can be understood by users when spoken by a voice device.

**Keywords:** Moodle LMS; Amazon Alexa; education; e-learning; Intelligent Personal Assistant; voice-content; usability

---

## 1. Introduction

Education is continuously challenged to achieve improvements on its learning processes and student performance [1]. The needs of learning have been changing over time, due to the rapid adoption of instant communication and the increase of information availability [2]. New teaching and learning methodologies have drastically altered education during the twenty-first century [3], with the integration of technological resources such as virtual platforms and hypermedia resources within the teaching-learning process, as well as other methodological techniques such as project-based or problem-based learning [4,5].

Pedagogy supporting deeper learning, which involves collaborative, informed, and personalized learning [6], has been determined to be more successful when helping students in achieving a more detailed understanding of modern skills [7]. The core will be a combination of personalization, collaboration, and informalization (informal learning), with traditional institutions finding the need to experiment with new formats and strategies for educating, providing them with the possibilities to offer relevant, effective and high-quality learning experiences [8]. However, while it has been identified improvement towards the student outcome brought by the use of collaborative learning [9,10],

additional factors have to be considered by teachers and instructors to achieve an effective and guaranteed learning [11,12].

Learning Management Systems (LMS) are considered a critical and decisive key of modern e-learning [13–15], and while sometimes they are regarded as just a tool to deliver content, LMSs provide the means to create engagement between the students, the instructors, and the content [16]. There is a continuous increase on the frequency that teachers and students interact through LMSs [17], and even though institutions have customized their own LMS instances to meet the instructors' needs [18], recent studies highlight the challenges presented in current LMSs to meet the specific needs of the student, regarding usability and learnability [19,20], which can cause a decline on their engagement with the platform and impact their learning outcomes [21,22].

The need to innovate and experiment with cutting-edge technology provided the opportunity and motivation to develop an Alexa skill to support the research effectiveness of utilizing a VUI for education, delivering notifications and upcoming events related to student's courses, and study if, and how, VUIs could be used as an educational tool to enhance the student's experience [23]. To such end, the Alexa skill would need the capability to integrate with Moodle. Furthermore, the Alexa skill would be developed to a production-ready level, to be released as soon as it is ready, and become publicly available to any user of Amazon Alexa. Finally, as the relevance of a robust architecture for IPAs is trending with several IPA architectures proposals in recent years [24], this project would aim as well to validate the existing knowledge available from scientific studies of similar nature, testing that such knowledge helps in our development to overcome issues we might encounter, and provides us with guidelines to implement a solid foundation for our application. Our work and findings would then be presented to provide new scientific insight for helping and guiding researchers on future works of similar characteristics, disclosing the successful and failed decisions and technical implementations taken along the way, along with the analysis and results of the usability of the skill itself.

Based on the research and motivation noted above, the research questions in this study are as follows. (RQ1) To build the basic architecture for an Alexa skill for educational purposes, including its integration with Moodle; (RQ2) to establish whether Moodle currently provides the necessary tools for voice-content creation for developing voice-first applications.

The remainder of this work is structured as follows. First, Section 2 presents a brief review of related work, followed by Section 3 which contains the methodology for our research and the implementation and development of the Alexa skill. Next, Section 4 describes the usability evaluation done after releasing the Alexa skill publicly, and Section 5 contains the analysis and results of our work. Finally, Section 6 discusses the main conclusions and limitations derived from our research.

## 2. Related Work

After an extensive search within multiple scientific databases (Scopus, Web of Science, ScienceDirect, Springer, Wiley, and Taylor & Francis), selected articles were reviewed in order to assess the state-of-art in research on the use of Voice User Interfaces for education. We divided the review of the found literature by first looking at the studies where there was no use of a smart speaker, followed by a review of those studies which did have some use of smart speakers. For this division, the concept of smart speaker was defined as a device which provides a way for people to interact with them by the use of their voice, without the needs to touch the actual device [25]. As well we created a comparison table to help us find commonalities between the studies, identifying the technical components, platforms, frameworks, tools, and libraries used on each of them (see Table 1).

**Table 1.** Comparison of studies.

Article	Voice Platform	LMS	Development Tools
Grujic et al. [26]	Interactive Voice Response (IVR)	Moodle	-
Pereira et al. [27]	Telegram app	open-MOOC	PHP, MYSQL
Todorov et al. [28]	Android app	DeLC 2.0	Java, CMUSphinx [29]
Kita et al. [30]	Google Simulator	Moodle	PHP, DialogFlow [31]
Kita et al. [32]	Google Home / Amazon Alexa	Moodle	PHP, DialogFlow, Alexa Cloud service
Laeq and Memon [33]	-	Moodle	Java, JADE (Java Agent Development Framework) [34]
Melton [35]	Amazon Alexa	Moodle	LAMP framework, Let's Encrypt

### 2.1. Studies with no Smart Speakers Use

In his work, Grujic et al. [26] propose a solution which links an E-education system, such as Moodle, via Interactive Voice Response (IVR), a technology that enables a computer to interact with humans through the use of voice and tones input produced by a keypad. The use of IVR applications could prove to be very beneficial, allowing the extension of operational business hours, reducing the waiting time for callers, and providing service to multiple users simultaneously, which would enhance service quality and the overall student's satisfaction. This study is a straightforward specification on how the prototype was implemented, and so there are no conclusions or outcomes we could use on our work regarding the impacts that such a prototype has on the users.

A solution for peer-to-peer (P2P) assessment using voice recordings is researched by Pereira et al. [27]. The authors implemented a chatbot using an existing Instant Messaging (IM) platform (Telegram), which integrated with a massive open online course (MOOC). Using the API provided by Telegram, the chatbot was developed to interact in a conversation with the student. There could be two possible conversation paths. On a normal interaction, the chatbot would ask a question to the user via text message. The user would then answer using the microphone capabilities of Telegram app, and the audio would be sent back to the chatbot for further processing and assessment. The second conversation path would be triggered if the user would send the custom-made evaluation command. The chatbot would present the original text question and the given audio answer so that the user could listen to it and provide a grade (on a scale from 1 to 10).

A prototype implementation of a VUI is discussed in the work by Todorov et al. [28]. The Learning Intelligent System for Student Assistance or LISSA, a name given to the prototype, is a personal assistant whose goal is to monitor the performance of tasks related to students' curriculum, to assist them along the learning process. LISSA's prototype takes the form of an Android app, implemented using Java programming language, and interacts with the DeLC 2.0 educational portal, which is a software project for developing an environment to support virtual education [36]. For our work, we found it highly beneficial to learn that several problems were found during the implementation of the LISSA's prototype and gather knowledge on how those issues were overcome. During the development process, it was found that the process for voice recognition was highly complex, categorized by the authors as an "entire scientific area". Todorov et al. [28], noted as well that recognizing voice commands does require a large amount of resources from the device, which impacted the drain of the device battery, considered by the authors as a valuable resource.

Kita et al. [30] argue that VUIs are quickly becoming suitable for various practical purposes in daily life, in addition to a VUI being an effective and intuitive interface to be used by many people, and so they describe the development process of a prototype VUI aiming for Moodle Quizzes used for educational purposes. Before learners decide to enroll into a course, a Moodle Quiz could be used as a tool that allows learners to quickly get an understanding of what can be learnt from the course and an appreciation of the educational gain the learners will achieve after its completion. The prototype developed by the author was made available to a limited number of users through Google Home smart devices. The conclusions of the study suggest that VUIs should aim to speak using short phrases, and that if there is need for long ones, then the VUI should inform users of the total time first, to avoid users to wonder when might it end. For this reason, the study concludes that content from an LMS must not be used without previous editing, to keep the learning experience through the VUI at its best. Furthermore, the study notes that the use of VUI for educational purposes is expected to have a positive effect on the learners’ motivation due to the notion that learners “feel heard”.

### 2.2. Studies Using Smart Speakers

In a more recent study, Kita et al. [32] continue their work related to the use of VUIs for educational purposes. In their latest work, the authors develop a voice app which allows searching for documentation regarding Moodle functions, with the use of voice commands, and has availability for both Google Home and Amazon Alexa devices (see Figure 1). The authors argue that such app would be helpful due to the rather common need to browse for documentation about operations methods and functions from Moodle when creating teaching material. The conclusions of the study denote as highly valuable the exploration of using VUIs as a channel to consolidate the interactions between LMSs and users. Moreover, the study emphasizes the need for the consideration of the advantages of choosing and implementing VUIs as user interfaces, while being aware of the existing limitations and drawbacks. Furthermore, the conclusions tease the future plan to extend the voice app with the capability to notify users about important activities, such as overdue assignments or short-coming events.

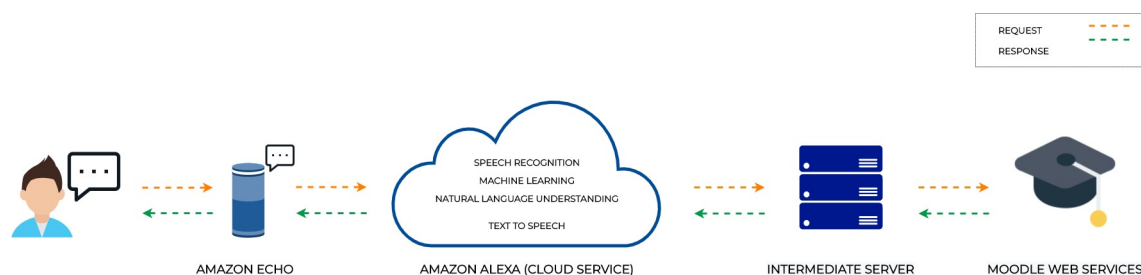
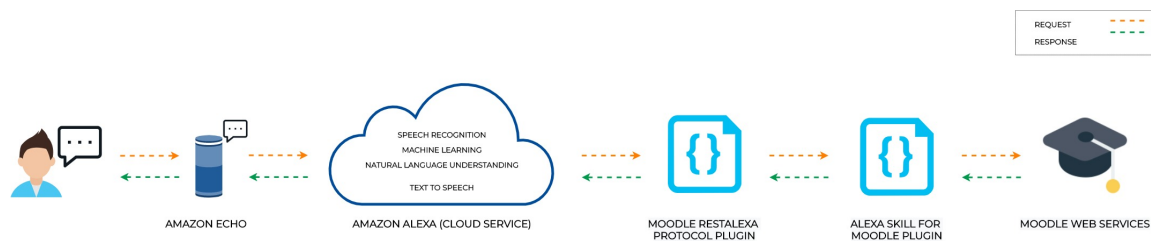


Figure 1. Graphical representation of Kita et al. [32] prototype architecture (Source: own authors’ elaboration).

In their study, Laeeq and Memon [33] define the usability of LMS systems as one of the key features that determine the success or failure of those systems, and relates to previous studies concluding that e-learning environments are not evolving according to the needs of learners, with poor usability as the core difficulty, produced by multiple design, navigation, and user interface issues found in existing LMS systems. Laeeq and Memon [33] reasons that navigations and search are two key modules of any LMS which impact system usability directly, and so a user-friendly navigational system within an LMS system can aid users to accomplish their goals, keeping them engaged until they have achieved their purpose. The conclusions from the works of Laeeq and Memon [33] indicate that the students who did interact with Moodle using the prototype built for the study show “greater motivation and a high rate of task completion in a shorter time period”, which presumes that integrating a voice-activated virtual assistant with an existing LMS has a distinguished and notable impact on students’ performance.

The authors' suggestion is to generalize the system in order to gather further data, as the usability of the prototype was tested within just a pre-selected group of students.

The work by Melton [35] describes the development of an Alexa skill that aims to enhance the speed and convenience of accessing information in Moodle. The author proposes the idea that an Alexa Skill would allow users to access information from LMS systems more quickly and conveniently, with immediate benefits for the users through site announcements to all users, course announcements to students and teachers, and overall course grades and upcoming due dates to students (see Figure 2). Besides improvements on the capabilities and features of the prototype itself, the conclusions of this work reveal areas for improvement regarding VUI design of the skill, mentioning that usability tests show a possible improvement in user experience by reducing responses to only provide information about the most recent events, shorting out the spoken response from Alexa, as opposed to provide with all available event information at once. Moreover, the interest of adding visual content to support spoken responses on newer Alexa devices which might incorporate graphic displays is noted, besides the already existing voice interface.

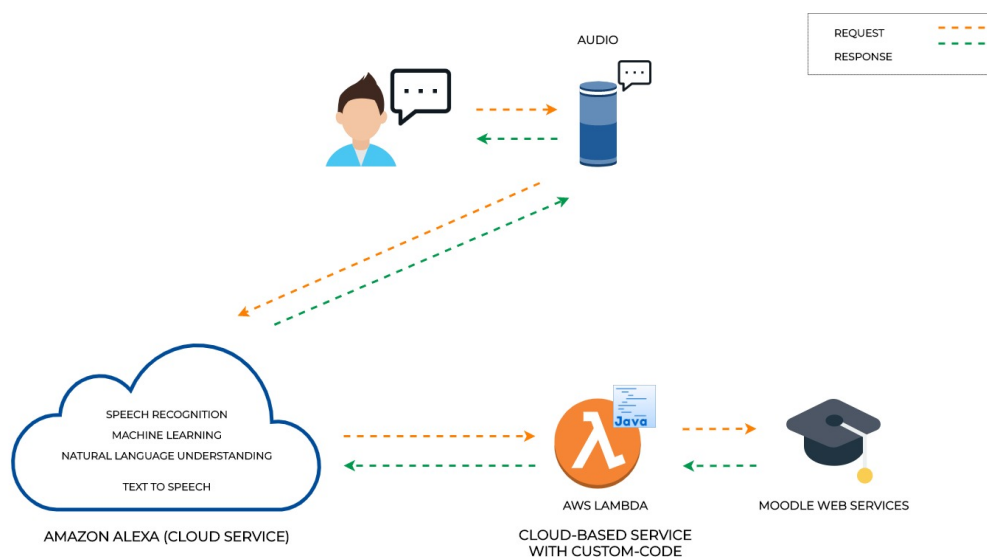


**Figure 2.** Graphical representation of Melton [35] prototype architecture (Source: own authors' elaboration).

### 3. Method

When a user speaks to a device with Alexa, the speech is streamed to the Alexa service in the cloud, which handles all speech recognition and conversion. Alexa service in the cloud processes the speech, determines what the user wants, recognizes if any of the available skills can fulfill the user's request and then sends a structured request to the particular skill [37]. A custom skill requires a cloud-based service to process the user's request and provide the proper response. As recommended by Amazon, the easiest way to build the cloud-based service for a custom Alexa skill is to use AWS Lambda. AWS Lambda is an Amazon Web Services (AWS) service that runs code only when it is needed and scales automatically, removing the need for provision or continuously running servers. The custom code for the Alexa skill is uploaded to AWS Lambda and the service does the rest, executing it in response to Alexa voice interactions and automatically managing the compute resources [37]. Additionally, due to the nature of the skill we implemented, our custom code required to be connected to a Moodle instance, as we had to take in use some of the functionalities provided by the Moodle Web Services.

The following sections explain more in detail the various modules that composed the final architecture of the custom Alexa skill we implemented for our research (see Figure 3).



**Figure 3.** Diagram for processing a request by our custom Alexa skill (Source: own authors' elaboration).

### 3.1. Moodle Web Services

For the implementation of our prototype we are required to connect programmatically to Moodle. To be more precise, our prototype took in use the REST protocol, which becomes available automatically when enabling the Mobile Web Services. Mobile Web Services is a built-in web service designed to enable the interaction between mobile applications and Moodle [38]. This web service is required for running the official Moodle Mobile app, or if a third-party app explicitly requires it. By utilizing the Mobile Web Services, we are capable of exploiting the same information as the official Moodle app uses to provide features such as browse the content of courses, track personal progress, or view course grades [38]. It is important to notice that the Moodle release used for our test case was "3.7.4+ (Build: 20200117)".

For our prototype, we initially took in use three core API functions:

- "core\_webservice\_get\_site\_info", which returns site info, user info, and list web service functions.
- `core_enrol_get_users_courses`, which gets a list of course ids that a given user is enrolled in.
- `core_calendar_get_calendar_events`, which gets calendar events for a given user.

We aimed to combine the use of the above-mentioned Moodle functions in order to retrieve enough information to present to a user with the specific events related to courses and other user-related matters. Later in this chapter, there is a more detailed explanation of how these functions were used and integrated into our Alexa skill.

### 3.2. Alexa Skill Implementation

First, we had to decide what type of Alexa skill we needed to build. Alexa provides a set of built-in capabilities, referred to as skills. Users can access these new abilities by asking Alexa questions or making requests. However, the functionality to be implemented determines how the skill integrates with the Alexa service and so it dictates what kind of skill needs to be built [37]. At the time we began building the Alexa skill there were existing pre-built skill models from where to choose (See Figure 4), but none of them provided us with the kind of interaction that we would need the skill to perform, and so we decided to develop a custom interaction model skill or "Custom Skill", which can handle just about any type of request [37].

Within the Alexa service, an interaction model is similar to what a graphical user interface is for a traditional app. Instead of clicking buttons and selecting options from dialog boxes and graphical elements, users make their requests and respond to questions by voice. When users speak questions and make requests, Alexa uses the interaction model to interpret and translate the words into a specific request that can be handled by a particular skill [37]. An extract of the Alexa skill's intent schema can be seen on Listing 1. Additionally, to build a custom skill we need an internet-accessible endpoint where our cloud-based service would be hosted. Alexa service is already integrated with AWS Lambda (an Amazon Web Service offering), so it makes it simple to host and run the service with our custom logic. Due to the already existing knowledge from the researchers, we decided to use this service and develop the cloud service for our skill using Java technology, specifically Java SE 8, which is supported by AWS Lambda [37].

Listing 1: Intent Schema used for our custom skill.

```
{
  "interactionModel": {
    "languageModel": {
      "invocationName": "calendario de la ubu",
      "intents": [
        {
          "name": "AMAZON.CancelIntent",
          "samples": []
        },
        {
          "name": "AMAZON.HelpIntent",
          "samples": []
        },
        {
          "name": "AMAZON.StopIntent",
          "samples": []
        },
        {
          "name": "AMAZON.NavigateHomeIntent",
          "samples": []
        },
        {
          "name": "AMAZON.FallbackIntent",
          "samples": []
        },
        {
          "name": "CalendarIntent",
          "slots": [],
          "samples": [
            "Calendario"
          ]
        }
      ],
      "types": []
    }
  }
}
```

alex developer console

Create a new skill

Cancel Create skill

Model: Custom Host: Provision your own

Skill name

Enter skill name

Skill name must have at least 2 characters. (50/50 characters)

Default Language

English (US)

More languages can be added to your skill after creation.

1. Choose a model to add to your skill

There are many ways to start building a skill. You can design your own custom model or start with a pre-built model. Pre-built models are interaction models that contain a package of intents and utterances that you can add to your skill.

Custom **Select**

Design a unique experience for your users. A custom model enables you to create all of your skill's interactions.

Flash Briefing

Give users control of their news feed. This pre-built model lets users control what updates they listen to.

"Alexa, what's in the news?"

Smart Home

Give users control of their smart home devices. This pre-built model lets users turn off the lights and other devices without getting up.

"Alexa, turn on the kitchen lights?"

Music

Give users complete control of their music. This pre-built model lets users search, pause, skip, or shuffle in your skill.

"Alexa, play music by Lady Gaga"

Video

Let users find and consume video content. This pre-built model supports content searches and content suggestions.

"Alexa, play Interstellar"

Meetings

This pre-built model leverages Alexa for Business APIs to allow users to search for and book available meeting rooms in their office.

"Alexa, book a room"

2. Choose a method to host your skill's backend resources

You can provision your own backend resources or you can have Alexa host them for you. If you decide to have Alexa host your skill, you'll get access to our code editor, which will allow you to deploy code directly to AWS Lambda from the developer console.

Provision your own **Select**

Provision your own endpoint and backend resources for your skill. This is recommended for skills that have significant data transfer requirements. You will not gain access to the console's code editor.

Alexa-Hosted (Node.js)

Alexa will host skills in your account up to the AWS Free Tier limits and get you started with a Node.js template. You will gain access to an AWS Lambda endpoint, 5 GB of media storage with 15 GB of monthly data transfer, and a table for session persistence. [Learn more](#)

Alexa-Hosted (Python)

Alexa will host skills in your account up to the AWS Free Tier limits and get you started with a Python template. You will gain access to an AWS Lambda endpoint, 5 GB of media storage with 15 GB of monthly data transfer, and a table for session persistence. [Learn more](#)

© English (US) © 2019 - 2020, Amazon.com, Inc. or its affiliates. All Rights Reserved. Terms Docs Forums Blog Alexa Developer Home

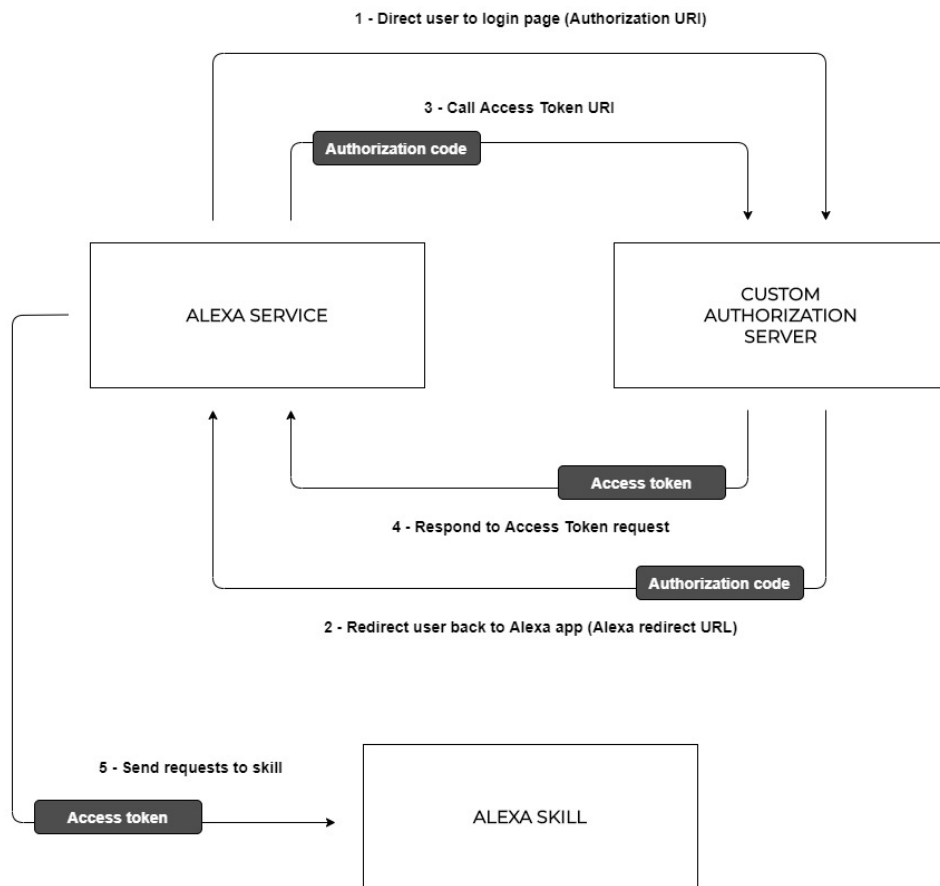
Figure 4. Alexa Developer Console: View for creating a new skill.

### 3.2.1. Account Linking Web Service

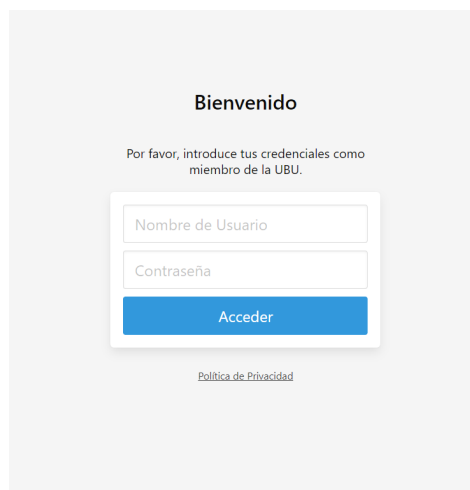
We required an additional web service that would handle the Account Linking process with Alexa service when users enable the skill for the first time. In a nutshell, Account Linking is an Alexa service's feature that enables authentication with a third-party service, and it is based on the OAuth2 authentication protocol (see Perez [39] for more details on OAuth2 protocol). The process for linking an account is initiated from the Alexa mobile app, which redirects the user to an authorization url, previously configured on the Alexa Developer Console (see Figure 5). Users would have to sign in to the form presented on that url to authenticate themselves against the third party service. In case the authentication is successful, a call to the access token url is triggered to fetch an access token to accompany any further Alexa request, and the Account Linking process is concluded. Similar to the authorization url, the access token url must be previously configured [40]. Account Linking is not mandatory for an Alexa skill to function properly, but in our case we perceived it would be best to add the extra step of linking the Moodle account of the user before any interaction with the skill happened, and so we implemented this service using Java SE 8 and AWS Lambda as we did previously.

Our skill aimed to retrieve information from Moodle LMS, which made it mandatory for the skill to communicate with Moodle Web Services. The Alexa requests would have to carry the account token to successfully interact with the Moodle Web Services. To reach the token, our approach was to develop a traditional login web form (see Figure 6) where the user could input the username and password of their Moodle account and exchange those credentials for the token. To keep the highest security when handling the user's password for Moodle, we decided not to store the password at all. The user would input the username and password from the Moodle account on the login form and those credentials would be sent directly to Moodle via HTTPS request to the script located at `/login/token.php`, which will then exchange those credentials for the account token, to be used in the following requests to Moodle Web Services.





**Figure 5.** Account Linking flow for authorization code grant type (Source: own authors’ elaboration).



**Figure 6.** Login web form screenshot (Source: own authors’ elaboration).

All communication was achieved using HTTPS to ensure the protection of the privacy and integrity of the exchanged data while in transit. As happened with the passwords, we decided not to store the Moodle token anywhere either. However, we would require the token when requesting information to Moodle Web Services. To be able to keep the token without storing it ourselves, we decided to use the Alexa access token for this task (see Figure 7).

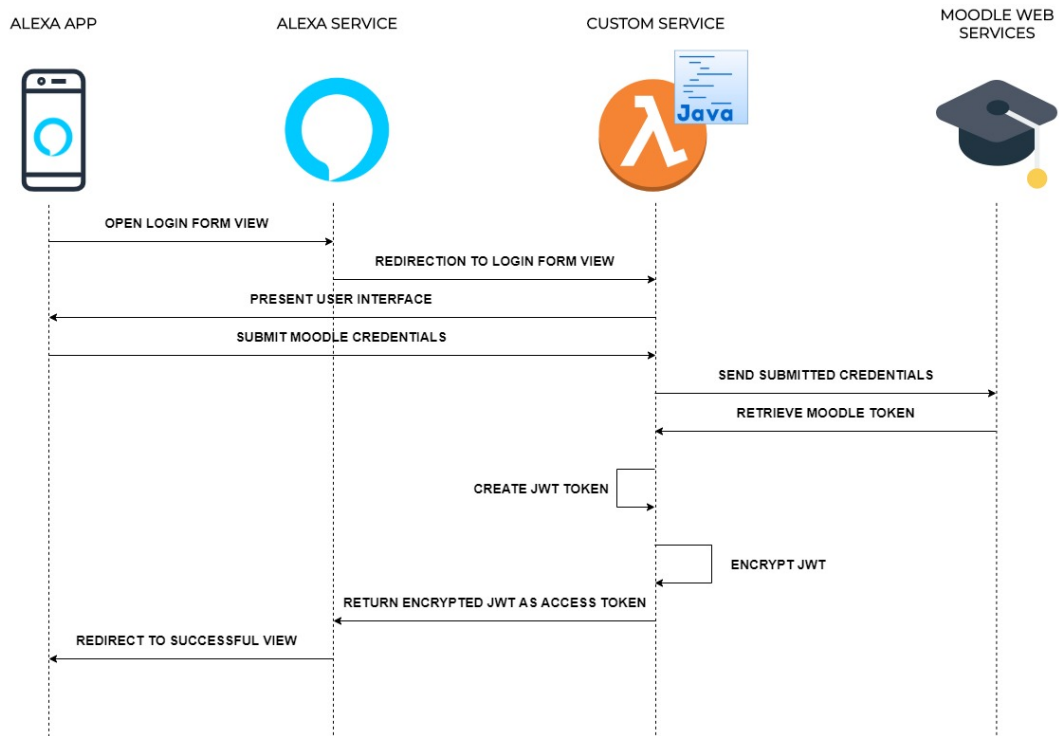


Figure 7. Flow diagram for Alexa access token exchange (Source: own authors’ elaboration).

The Alexa access token is delivered on every Alexa request sent by the Alexa service, and allows for identifying the user which triggered the request. We took advantage of this feature and used the Alexa access token to store a JSON Web Token (JWT) created by us. The JWT would contain information that would allow us to identify the user (see Listing 2), including the Moodle account token previously exchanged through the login form. JWTs are an open industry standard RFC 7519 method for representing claims securely between two parties. As explained on the RFC formal document by Jones et al. [41], “the claims in a JWT are encoded as a JSON object that is used as the payload of a JSON Web Signature (JWS) structure or as the plaintext of a JSON Web Encryption (JWE) structure, enabling the claims to be digitally signed or integrity protected with a Message Authentication Code (MAC) and/or encrypted”.

Our JWT token would include three claims for us to use later on:

- **userId**, our own unique identifier for the user, not related to any identifier from Alexa service or Moodle.
- **username**, user’s Moodle username.
- **moodleToken**, the token retrieved from Moodle after user’s authentication through the login form.

With the implementation of the JWT token we were able to identify the user when needed, as well as being capable to interact with Moodle Web Services on user’s behalf. However, we felt that passing the JWT token as plain text might be insecure due to the fact that the Moodle token of the user would be readable to any third party that might gain access to the JWT token. For this reason, we decided to encrypt the JWT token before passing it through to Alexa service. The main objective of cryptography is to keep data safe from unauthorized access.

## Listing 2: JWT structure example.

```

Header: Algorithms and Token Types
{
  "alg": "HS256",
  "typ": "JWT"
}

Payload: Data
{
  "iss": "XXX",           // Public Claim
  "iat": XXX,           // Public Claim
  "userId": "XXX",      // Private Claim
  "username": "XXX",    // Private Claim
  "moodleToken": "XXX", // Private Claim
}

Signature: results from Hash
{
  HMACSHA256(
    base64UrlEncode(header) + "." +
    base64UrlEncode(payload),
    secret)
}

```

Through cryptography we can achieve security by encoding messages in unreadable format so that the original message would only be read by the intended user [42]. While there are many encryption mechanisms currently available, we decided to use “Threefish” to securely encrypt the JWT token. “Threefish” is a large, tweakable block cipher, developed by Bruce Schneier, Niels Ferguson, Stefan Lucks, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, and Jesse Walker in 2008 [43]. It is defined for three different block sizes: 256 bits, 512 bits, and 1024 bits. We used 512 bits as the block size to encrypt the JWT token. The key is the same size as the block, and the tweak value is 128 bits for all block sizes [44].

Finally, we encoded the encrypted token using Base64 algorithm. The Base64 algorithm allows for encoding and decoding an object into ASCII format, and it is used in multiple cases, including URL encoding purposes [45]. By encoding the encrypted token we made sure the data integrity would remain during the sending and receiving of the token. The final encoded token is then ready to be sent back to Alexa service and complete the Account Linking process successfully. The encoded token would be then sent as part of any future Alexa request. Receiving the encoded token within the requests would allow us to identify the user who sent the request, and successfully communicate with Moodle Web Services on user’s behalf, as the Moodle account token would be contained within the JWT as claim, once it is decoded and decrypted.

### 3.2.2. Implementation Problems and Workarounds

Over the course of the development of the skill we confronted unexpected issues regarding the user experience when creating voice dialog from the information we got from Moodle Web Services. As mentioned in Section 3.1, we combine several Moodle functions to create the logic that would reach the calendar event data for the user. We utilize the function `core_enrol_get_users_courses` to retrieve the information regarding the course, such as the name of the course, and the function `core_calendar_get_calendar_events` to retrieve the actual information related to the events, such as name of the event, date of the event, etc.

We encountered the problem that certain properties of the elements of Moodle, such as the name of a course or the name of an event, might be specified by the administrator of such element in such manner that would not be directly compatible with audio streaming. For example, some event titles might contain special characters to help understand that the event is aimed for a specific group

of people (e.g., Project Presentation (Group 1) or Basics of Programming (Online)). Furthermore, course titles might be too long for an audio experience, where the user must listen to the full course name if events related are presented, which might exhaust the time and willingness of the user to listen to the full response for a long period of time. This problem with the user experience brought up a new issue, this time regarding the administration of the Alexa skill itself. How could an administrator of an element in Moodle, such as a calendar event, for example, be able to hear what the final user might experience through the Alexa devices, before making it publicly accessible? As an example, if a teacher could listen to what the students might hear regarding an event that is being newly added, the teacher is still able to edit the outcome until the audio experience achieves the expected level of quality and comprehension. However, in its current form, the only way for any person to hear what an event might sound like through our Alexa skill was to actually make the event public, without even knowing how it will really be spoken out by the Alexa device.

These problems were a direct risk for the purposes of our work. If we were to test the likelihood of users enjoying the use of our Alexa skill, we had to aim for the best user experience we could offer to begin with. As our work was not a technical research on Moodle integration with Amazon Alexa, we made the conscious decision to edit the interaction flow with Moodle. Instead using the Moodle function `core_calendar_get_calendar_events` to fetch the user events, we decided to create a file within our application which would contain the events information hard-coded within, and replace the use of the Moodle function by reading and processing that file when needed. By hard-coding the information on a file, we removed the flexibility of updating event data through Moodle, or any external service for that matter. The only way to update any event data, meaning adding new events, deleting unnecessary ones, or modifying any of the existing ones, could only be achieved by editing the file manually and deploying the application again on our cloud hosting service (in our case AWS Lambda service). On the other hand, this file contained all the necessary information we required for our skill to perform optimally. The file would contain event data in JavaScript Object Notation (JSON) format, which will allow us to have custom values such as the course name, as we wanted to be said by Alexa, the date of the event, on a standard ISO-8601, and other values what allowed us to build the audio output of the Alexa skill on a more user-friendly fashion, which would help to our research for understanding user's experience with Alexa (see example of event JSON on Listing 3).

Listing 3: Calendar event JSON example.

```
[
  {
    ...
  },
  {
    "eventId": "002",
    "eventTitle": "Final day for Early Stimulation course",
    "eventDate": "2020-06-26",
    "eventCourseId": "1685"
  },
  {
    ...
  }
]
```

#### 4. Usability Evaluation

We confronted several issues while developing our Alexa skill. One of the conscious decisions we made was to hard-code part of the events' information on a file, as opposed to get the information programmatically from Moodle instance directly. This decision meant that, while any user with an account in Moodle would be able to access and interact with the Alexa skill, only those who were participants on the particular courses we had chosen and which events' information was included in

the file. However, as mentioned, any user with an account on Moodle would be able to access and interact with the Alexa skill, and that would include the introduction to the use of the skill, as well as the account linking process and other available paths of interactions.

Once our skill was fully developed, deployed, and publicly available, we began a usability evaluation to understand the performance and impact that the skill could have on the students. It is of importance to note that we did not recruit any student to participate in the evaluation of the skill. The students were told about the Alexa skill on the course presentation day and the goals of the project were explained to them. As part of the evaluation we aim to measure what would be the acceptance level of the Alexa skill and how students would adopt such innovation, understanding if students would interact with the skill willingly on their own. To increase the amounts of data measured over the evaluation period, we decided to boost the interest of students for using the skill on their own by offering an Alexa Echo Flex device as reward from a drawing between those students who would interact with the skill.

Totally, there were 61 students accounted to be part of the course to be used for the usability evaluation. Prior the beginning of this study, both the authorization of the Bioethics Committee of the University of Burgos and the informed consent of all participants were obtained in writing (see the Ethics Statement section at the end of this article). The student group was composed by 56 female and 5 male students, with further statistical data to be seen on Table 2.

**Table 2.** Description of the evaluation sample.

Participant Gender	Count	Mean Age	Standard Deviation Age
Male	5	22.00	2.24
Female	56	23.54	6.30
Total	61	23.36	6.01

The evaluation was conducted over the period of two months, from 1 February 2020 until 31 March 2020. Information regarding the usage of the skill was taken from the Analytics section found on the Alexa Developer Console (See Figure 8). This tool makes it possible to monitor many aspects on how the skill is being utilized by the users and how it is performing. These measurements will allow us to study if, and how, the technical performance of the skill might have a positive or negative impact on its usability, and detect those areas where problems might arise during the interaction of the users with the skill.

A short non-mandatory survey was given to students at the beginning of the course to better understand their familiarity with Amazon Alexa technology. This survey was composed of one question with dichotomous reply options (yes/no) and a second one with multi-choice question (see Table 3). The answers from this survey offered us information to create participant profiles, which could later on help us understand better the results obtained by the usage of the skill and the usability evaluation (see Figure 9). From the 26 students who participated in the survey, 12 of them (46.2%) answered that they had used a voice-activated device before this evaluation. Only one of the participants answered to use the device on a weekly basis, and ten of them answered to use it seldom.

**Table 3.** Initial survey to students (Extract taken from Saiz-Manzanares et al. [23]).

Survey Question	Possible Answers
Have you ever used a voice-activated device, similar to Alexa Echo devices?	Yes No Never Seldom
If so, how often do you use those devices?	Weekly Several times per week Daily

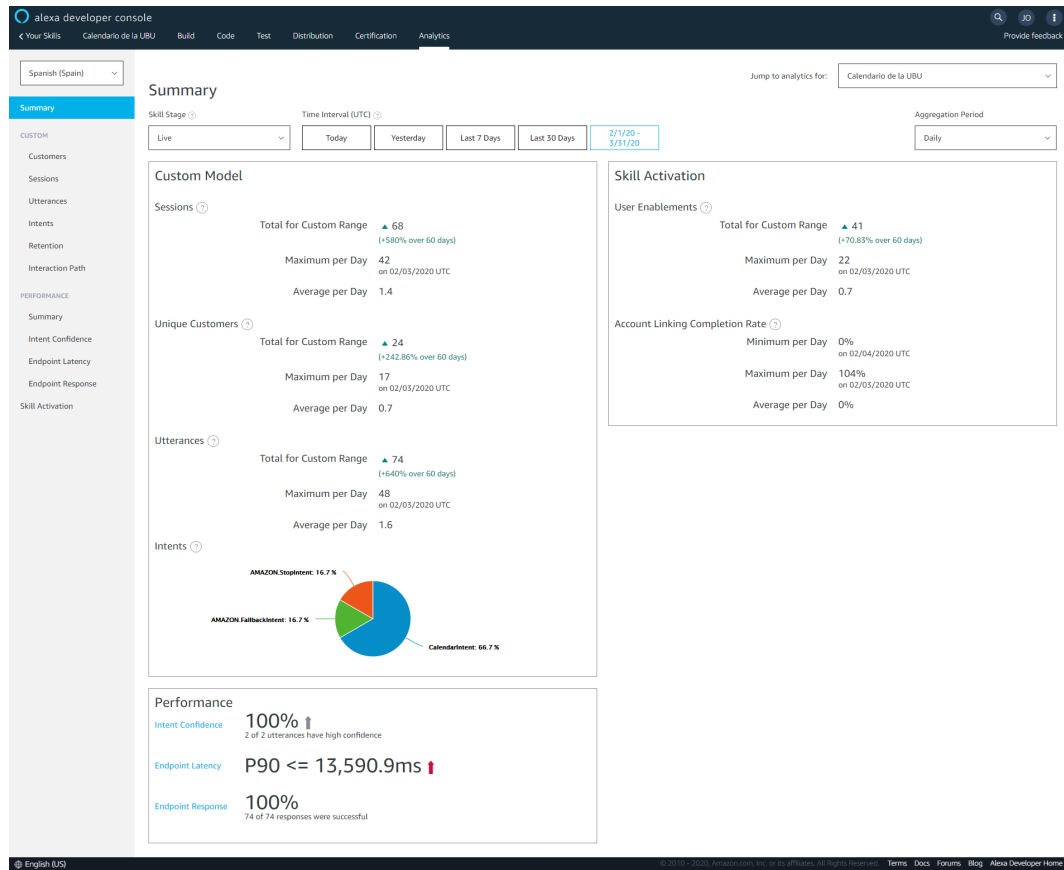


Figure 8. Screenshot of the Analytics section within the Alexa Developer Console.

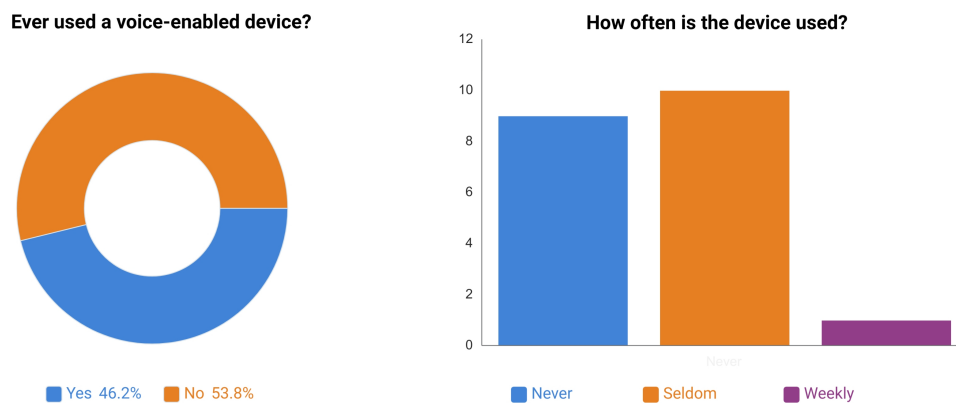


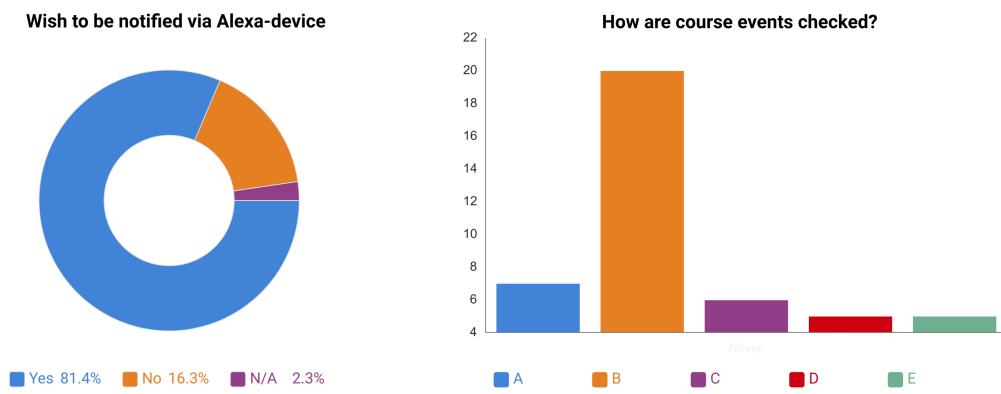
Figure 9. Statistical data deviated from initial survey (Source: own authors’ elaboration).

Finally, a feedback survey was designed and given to the students at the end of the course. This survey consisted of one multi-choice question and a second one with dichotomous reply options (yes/no), with three more open-ended questions. We did not implement reliability analysis as the survey aimed to get a qualitative opinion from the students (see Table 4). We gathered answers from 43 students (70.5% from the total amount of participants). The feedback given to us by these students allowed us to better understand their personal experience when interacting with the Alexa skill, as well to collect information on how to further improve it. One of the most interesting outcomes of the survey was to find out that a vast majority of the students (81.4%) would like to receive events through an Alexa-enabled device, but only 6 of the surveyed students (13.9%) did answer they used the Alexa skill to do so (See Figure 10).

Students would like to have more content on the current Alexa skill, which can enable them to get information regarding all their courses (different activities within the course, exam dates, material release dates, essay deadlines, etc.), as well as other information regarding the school, such as upcoming special events (expositions, talks and conferences, etc.). In contrast, the survey indicates that Moodle is seen as a very complete and useful platform as is. Finally, the answers to the survey suggest that students are not as familiarized with voice technology as with other current technologies, as, for example, mobile apps, and voice technology is not yet integrated in their daily lives. This unfamiliarity is reflected in the existence of a certain level of insecurity from the students when it comes to utilizing technology, which in the case of Alexa devices, requires listening to the user continuously in order to activate and initiate the interaction with the user.

**Table 4.** Feedback survey to students (Extract taken from Saiz-Manzanares et al. [23]).

Survey Question	Possible Answers
What option do you use to know about events from the course?	A—Event’s calendar from Moodle B—Uploaded PDF from the teacher C—Alexa skill D—Ask to other students E—I took notes of the dates given the first day of class
Do you like to receive notifications through Alexa-enabled devices?	Yes No
What additional information would you like to receive through the Alexa skill?	Open answer
What additional information would you like to receive through Moodle platform?	Open answer
If you did not use the Alexa skill, please tell us more about why and give us suggestions for improvement	Open answer



**Figure 10.** Statistical data deviated from the feedback survey (Source: own authors’ elaboration).

### 5. Results

Once the usability evaluation was completed, we moved to further investigate the results obtained by the usage of the Alexa skill, and compare such results with the feedback previously gathered from the students. The usage of the skill was relatively low. On February 3rd, the first day of the course, the Alexa skill was introduced to the students. It is during that brief time of introduction (around 15 min) when most of the interaction with the Alexa skill happened. The usage of the Alexa skill during the time of evaluation can be seen on Figure 11. There are some sporadic interactions with the skill during the following days and weeks. However, those sporadic interactions occurred on the same days as the instructor gave in-class reminders about the Alexa skill.

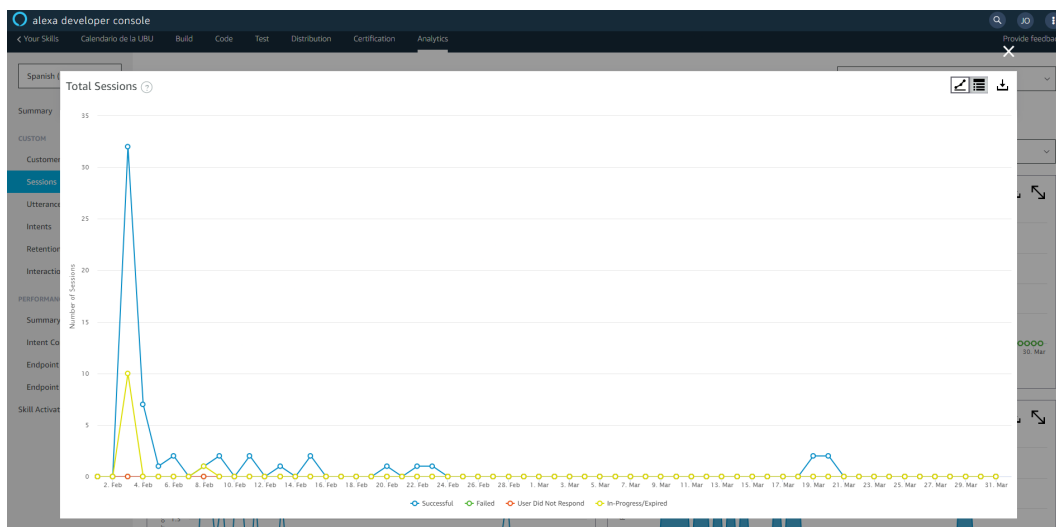


Figure 11. Total sessions opened towards the Alexa skill.

We aimed to investigate if there could be some technical issue that would be affecting the usability experience of the skill. We began by looking at the skill activation and account linking processes. We found out that, while most of the students linked their accounts successfully during the presentation day, all the intents in following days were never completed successfully (see Figure 12). We found no record of error related to the activation or account linking processes on the skill logs, and the Alexa Developer Console did as well concur on the fact that there were no errors to be found. However, the feedback survey did provide us with a possible explanation.

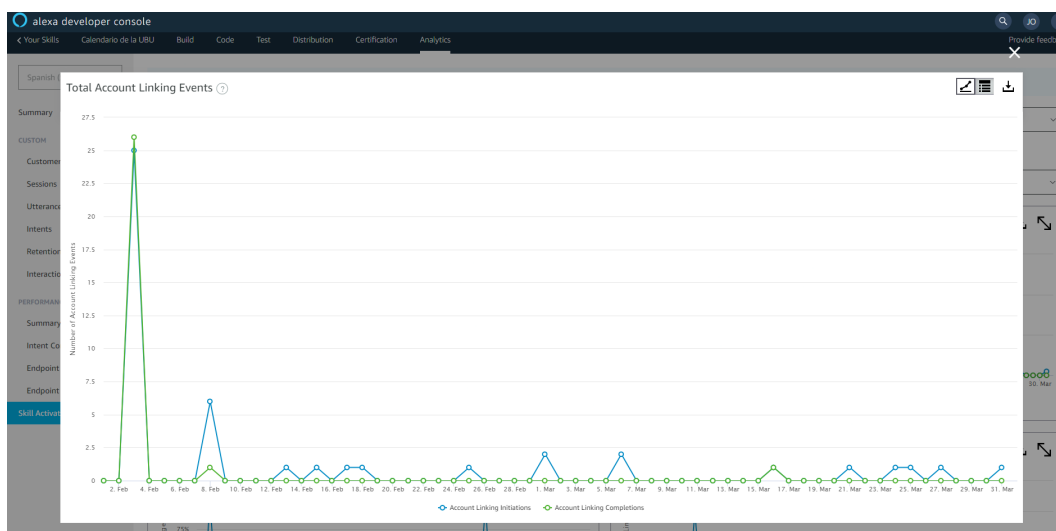


Figure 12. Account Linking data for Alexa skill.

It is worth mentioning that the results of the feedback survey are further detailed and exhaustively analyzed in the works by Saiz-Manzanares et al. [23], where a more profound study on the effectiveness of using IPAs in education is performed. This particular study analyzes what is the students’ perception on the usefulness of IPAs when used in conjunction with LMSs, researching if there are significant differences in how students access to information provided by LMSs, as well as if there are significant differences in the students’ learning outcomes, depending on use versus non-use of an IPA. From the students’ feedback we could deduce that providing access for the skill to their private data located in Moodle might be too much to ask from them in proportion to the gain. They felt the Alexa skill would not provide them with value enough to equal for the potential privacy risks. Additionally, these results match the expected outcome given the participant profiles we defined from the evaluation’s initial



survey. More than half the students that participated on the survey had never used a voice-enabled device, with some of them using them seldom, if ever. It would be expected that users would be reluctant to go over an account linking process when they are not comfortable utilizing the device itself as part of their daily routine.

Furthermore, from the feedback survey, we can extract the sense of completeness that Moodle brings to the students. The feedback gained on possible improvements for the skill reflects the expectations from the students for the Alexa skill to be able to operate at the same level as Moodle, providing the same wide range of information related to not only courses but any kind of event regarding the educational day-by-day life of the student. Similar to the activation and account linking processes, we found no evidence of error that might cause users to stop using the skill or which would impact negatively to the experience. As previously mentioned, most of the interactions with the skill happened during the presentation day, with sporadic usages on the same days as the instructor reminded the class about the skill. Once again, based on this insight and the data gathered from the feedback survey, we can resolve that students are not yet familiarized with voice technology and do not feel comfortable as to integrating it on their daily routines.

Easy and fast accessibility to content are highly valued needs for students, and platforms like Moodle already provide a solution for such expectations, to a certain degree, while to really match such expectations, Moodle and voice technology require to attain more elaborated integration capabilities than currently available. Today, Moodle offers powerful tools to gather information from external applications, such as the Moodle Web Services. However, in order to create smooth and realistic interactions between users and Alexa devices, there is a need for a missing layer of interoperability.

As we found during the development process, the information gathered from Moodle via Web Services required to be processed so that Alexa could dictate the sentence in a way that the user could make sense out of it. We were forced to implement this missing layer, which required to hard-code into our application part of the data that would be necessary for the construction of those sentences to be spoken by the Alexa device. As the data had to be hard-coded, only developers were able to influence and modify the resultant outspoken sentence, but people with no software development knowledge, as, for example, some of the course instructors, could not make any changes. While this is a manageable situation when developing a prototype for close-controlled tests, it is not a plausible alternative for a production-ready application. Instructors must be able to manage information within Moodle to create voice-specific content through the same user interfaces used currently to create visual content displayed on the web. Moodle would have to be enhanced with new forms of inputting information for voice-specific matters, in addition to new tools that will allow instructors to pre-listen the outspoken sentences and do the necessary adjustments, prior to publicly releasing such content. It is when the instructor feels comfortable with the tools to create such voice content that the student would receive a gratifying experience when interacting with voice devices, which then would lead to an increase of the interaction with the voice device, and ultimately making the interaction with voice devices an integral part of the user's educational life.

## 6. Conclusions and Future Work

In this paper, we present the work done to build an Alexa skill for educational purposes, including its integration with Moodle, as response to our research question (RQ1), the development of which takes into consideration the existing scientific knowledge currently available. Our aspiration is to provide new scientific insight to help and guide researchers on future works of similar characteristics by exposing our successes and failures from the decisions and technical implementations taken along the development process.

Contrary to the common approach taken by previous research on the field, we aimed to build our prototype of Alexa skill so that it would be immediately released and publicly accessible, while the current research available provides studies where prototypes are tested under controlled circumstances, mentioning conclusions about the future possibilities of releasing a production-ready version [33,35].

Nevertheless, we carried out a usability evaluation as well, to provide us with insight regarding the usage of the skill. With an initial survey, we were able to create participant profiles for better understanding of the feedback provided by the students in the following surveys and to further comprehend the results gathered from the analytical data extracted from the usage of the skill.

For our research question (RQ2), and contrasting with some of the results we encountered on papers published in recent years, our findings indicate that Moodle is not a platform which currently provides the necessary tools for voice-content creation. Our findings indicate that, in order to create a voice experience with the use of Moodle data, there needs to be a middle layer that would process the data and produce sensible sentences that can be comprehend by users when spoken by a voice device. An alternative to build such missing layer of interoperability would be to implement enhancements that would allow Moodle to be ready for voice-content creation. As future lines of work, our team is considering how such enhancements could be brought to Moodle, either in the form of a Moodle plugin or as an additional application, similar to the Moodle mobile app, but for voice devices. Moreover, creating an additional platform that would bridge the gap between Moodle and voice devices could be an alternative.

Besides voice-content creation, usability can be improved as well, providing a better experience for students when interacting with Moodle via voice instead of traditional visual interfaces. A new line of future work opens for researching how to improve the Moodle platform to be used via voice. A possible path could be to introduce a voice assistant within the Moodle mobile app, or perhaps create new plugins for existing web browsers to interact with Moodle via voice directly from the computer.

Finally, we acknowledge the limitations of our Alexa skill, and we will continue working on functionality improvements in the upcoming developments of the skill. To enhance the limited set of commands understood by the skill, we aim to improve the skill to respond to any inquiry related to a course with no restrictions to what course that might be. Furthermore, the introduction of additional languages is on the scope for improvements. Currently the skill is only available in Spanish, but English could be very helpful for international students. Furthermore, we recognize the need to improve the methodology used when surveying participants regarding the experimentation and usability of innovative technology such as IPAs. For future studies, we perceive employing measurement tools as being beneficial for evaluation, for example, a System Usability Scale (SUS) [46] or User Experience Questionnaire (UEQ) [47].

## 7. Patents

UBUVoiceAssistant Computer application is registered in the Spanish General Property Registry of the Ministry of Culture and Sports; Registration number: BU-69-20 [48].

**Author Contributions:** Conceptualization, M.C.S.-M., R.M.-S. and J.O.-O.; methodology, J.O.-O.; software, J.O.-O.; validation, M.C.S.-M., R.M.-S.; formal analysis, M.C.S.-M., R.M.-S.; investigation, M.C.S.-M., R.M.-S. and J.O.-O.; resources, M.C.S.-M., R.M.-S. and J.O.-O.; data curation, M.C.S.-M., R.M.-S.; writing—original draft preparation, 493 J.O.-O.; writing—review and editing, M.C.S.-M., R.M.-S. and J.O.-O.; visualization, J.O.-O.; supervision, M.C.S.-M., 494 R.M.-S.; project administration, M.C.S.-M., R.M.-S. and J.O.-O. All authors have read and agreed to the published version of the manuscript.

**Funding:** The grant from the Vice-Rectorate of Teaching and Research Staff, the Vice-Rectorate for Research and Knowledge Transfer, 2020, at the University of Burgos (Spain).

**Acknowledgments:** Thanks to all the students who participated in this study and the Committee of Bioethics of University of Burgos (Spain).

**Conflicts of Interest:** The authors declare no conflict of interest.

**Ethics Statement:** The Ethics Committee of the University of Burgos approved this study, N° IR 30/2019. Written informed consent was in each case requested from the students who participated in this research. They all gave their written informed consent in accordance with the Declaration of Helsinki.

## References

1. Saiz Manzanares, M.C.; Cuesta Segura, I.I.; Alegre Calderon, J.M.; Peñacoba Antona, L. Effects of Different Types of Rubric-based Feedback on Learning Outcomes. *Front. Educ.* **2017**, *2*, 34. [CrossRef]
2. Lapointe, J.F.; Molyneaux, H.; Kondratova, I.; Viejo, A.F. Learning and Performance Support—Personalization through Personal Assistant Technology. In *Learning and Collaboration Technologies*; Zaphiris, P., Ioannou, A., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 223–232. [CrossRef]
3. Cerezo, R.; Sánchez-Santillán, M.; Ruíz, M.; Nuñez, J. Students LMS Interaction Patterns and Their Relationship with Achievement: A Case Study in Higher Education. *Comput. Educ.* **2016**, *96*. [CrossRef]
4. Saiz Manzanares, M.; Llamazares, M.d.C.; Gonzalez, A. Effectiveness of Blended Learning in Nursing Education. *Int. J. Environ. Res. Public Health* **2020**, *17*, 1589. [CrossRef]
5. Allen, L.; Taylor, D. The role of the academic Library Information Specialist (LIS) in teaching and learning in the 21st century. *Inf. Discov. Deliv.* **2017**, *45*, 1–9. [CrossRef]
6. Gijsbers, G. The Future of Learning: A Foresight Study on New Ways to Learn New Skills for Future Jobs. *Eur. Foresight Platf. (EFP) Brief* **2012**. [CrossRef]
7. Kugemann, W.; Aceto, S.; Dondi, C.; Kastis, N. Inspiring Young People to Become Lifelong Learners in 2025, 2009. Learnovation Research Report. Available online: <https://hal.archives-ouvertes.fr/hal-00593007> (accessed on 12 April 2020).
8. Redecker, C.; Leis, M.; Leendertse, M.; Punie, Y.; Gijsbers, G.; Kirschner, P.; Stoyanov, S.; Hoogveld, B. The Future of Learning: Preparing for Change, 2011. Available online: <https://core.ac.uk/reader/38621986> (accessed on 14 April 2020).
9. Yamada, Y.; Hirakawa, M. A Case Study of Analyzing Logs of LMS in Flipped Classroom. In Proceedings of the 2015 IIAI 4th International Congress on Advanced Applied Informatics, Okayama, Japan, 12–16 July 2015; pp. 374–378. [CrossRef]
10. Järvelä, S.; Malmberg, J.; Koivuniemi, M. Recognizing Socially Shared Regulation by Using the Temporal Sequences of Online Chat and Logs in CSCL. *Learn. Instr.* **2016**, *42*, 1–11. [CrossRef]
11. Bernard, M.; Bachu, E. Enhancing the Metacognitive Skill of Novice Programmers through Collaborative Learning. *Intell. Syst. Ref. Libr.* **2015**, *76*, 277–298. [CrossRef]
12. de Raadt, M.; Watson, R.; Toleman, M. Teaching and Assessing Programming Strategies Explicitly. In *Proceedings of the Eleventh Australasian Conference on Computing Education—Volume 95*; Australian Computer Society Inc.: Darlinghurst, Australia, 2009; pp. 45–54.
13. Kyzy, Z.N.; Ismailova, R.; Ündar, H.D. Learning Management System Implementation: A Case Study in the Kyrgyz Republic. *Interact. Learn. Environ.* **2018**, *26*, 1010–1022. [CrossRef]
14. Iádel Puerto Paule Ruíz, M.; ález, M.R.G.; áñez-Santillán, M.S.; ón Pérez Pérez, J.R. The Procrastination Related Indicators in E-learning Platforms. *J. UCS* **2015**, *21*, 7–22. [CrossRef]
15. Morgado, L.; Paredes, H.; Fonseca, B.; Martins, P.; Almeida, A.; Vilela, A.; Peixinho, F.; Santos, A. A Bot Spooler Architecture to Integrate Virtual Worlds with E-learning Management Systems for Corporate Training. *J. Univers. Comput. Sci.* **2016**, *22*, 271–297. [CrossRef]
16. Harris, H.S.; Greer, M. Over, Under, or through: Design Strategies to Supplement the LMS and Enhance Interaction in Online Writing Courses. *Commun. Des. Q. Rev.* **2017**, *4*, 46–54. [CrossRef]
17. Saiz Manzanares, M.í.a.C.; Marticorena Sánchez, R.ú.l.; Díez Pastor, J.é.F.; García Osorio, C.é.s.I. Does the Use of Learning Management Systems with Hypermedia Mean Improved Student Learning Outcomes? *Front. Psychol.* **2019**, *10*, 88. [CrossRef] [PubMed]
18. Alturki, U.; Aldraiweesh, A.; Kinshuk, D. Evaluating the Usability and Accessibility of LMS “blackboard” at King Saud University. *Contemp. Issues Educ. Res. (CIER)* **2016**, *9*, 33. [CrossRef]
19. Bartolotta, J.; Newmark, J.; Bourelle, T. Engaging with Online Design: Undergraduate User-participants and the Practice-level Struggles of Usability Learning. *Commun. Des. Q. Rev.* **2018**, *5*, 63–72. [CrossRef]
20. Oliveira, P.C.D.; Cunha, C.J.C.d.A.; Nakayama, M.K. Learning Management Systems ( LMS ) and E-learning Management: An Integrative Review and Research Agenda. *JISTEM- Inf. Syst. Technol. Manag.* **2016**, *13*, 157–180. [CrossRef]
21. Son, J.; Kim, J.D.; Na, H.S.; Baik, D.K. A Social Learning Management System Supporting Feedback for Incorrect Answers Based on Social Network Services. *Educ. Technol. Soc.* **2016**, *19*, 245–257.

22. Dahlstrom, E.; Brooks, D.; Bichsel, J. *The Current Ecosystem of Learning Management Systems in Higher Education: Student, Faculty, and IT Perspectives*; Technical Report; EDUCAUSE Center for Analysis and Research: Louisville, CO, USA, 2014. [CrossRef]
23. Saiz-Manzanares, M.; Marticorena-Sánchez, R.; Ochoa-Orihuel, J. Effectiveness of Using Voice Assistants in Learning: A Study at the Time of COVID-19. *Int. J. Environ. Res. Public Health* **2020**, *17*, 5618. [CrossRef]
24. Silva, A.d.B.; Gomes, M.M.; da Costa, C.A.; Righi, R.d.R.; Victoria Barbosa, J.L.; Pessin, G.; De Doncker, G.; Federizzi, G. Intelligent personal assistants: A systematic literature review. *Expert Syst. Appl.* **2020**, *147*, 113193. [CrossRef]
25. Bentley, F.; Luvogt, C.; Silverman, M.; Wirasinghe, R.; White, B.; Lottridge, D. Understanding the Long-term Use of Smart Speaker Assistants. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **2018**, *2*. [CrossRef]
26. Grujic, D.; Milic, A.; Dadic, J.; Despotovic-Zrasic, M. Application of IVR in E-learning System. In Proceedings of the 20th Telecommunications Forum, Belgrade, Serbia, 20–22 November 2012; pp. 1472–1475. [CrossRef]
27. Pereira, J.; Fernandez-Raga, M.; Osuna-Acedo, S.; Roura-Redondo, M.; Almazan-Lopez, O.; Buldon-Olalla, A. Promoting Learners' Voice Productions Using Chatbots as a Tool for Improving the Learning Process in a MOOC. *Technol. Knowl. Learn.* **2019**, *24*, 545–565. [CrossRef]
28. Todorov, J.; Valkanov, V.; Stoyanov, S.; Daskalov, B.; Popchev, I.; Orozova, D. Personal Assistants in a Virtual Education Space. In *Practical Issues of Intelligent Innovations*; Springer International Publishing: Cham, Switzerland, 2018; pp. 131–153. [CrossRef]
29. Shmyrev, N. CMUSphinx Open Source Speech Recognition, 2020. Available online: <https://cmusphinx.github.io> (accessed on 14 April 2020).
30. Kita, T.; Nagaoka, C.; Hiraoka, N.; Suzuki, K.; Dougiamas, M. A Discussion on Effective Implementation and Prototyping of Voice User Interfaces for Learning Activities on Moodle. *Int. Conf. Comput. Support. Educ.* **2018**, *1*, 398–404. [CrossRef]
31. Dialogflow. Dialogflow | Google Cloud, 2020. Available online: <https://cloud.google.com/dialogflow> (accessed on 14 April 2020).
32. Kita, T.; Nagaoka, C.; Hiraoka, N.; Dougiamas, M. Implementation of Voice User Interfaces to Enhance Users' Activities on Moodle. In Proceedings of the 2019 4th International Conference on Information Technology: Encompassing Intelligent Technology and Innovation Towards the New Era of Human Life, Bangkok, Thailand, 24–25 October 2019; pp. 104–107. [CrossRef]
33. Laeeq, K.; Memon, Z. Scavenge: An Intelligent Multi-agent Based Voice-enabled Virtual Assistant for LMS. *Interact. Learn. Environ.* **2019**, 1–19. [CrossRef]
34. Bellifemine, F.; Poggi, A.; Rimassa, G. JADE—A FIPA -Compliant Agent Framework, 1999. Available online: <http://jmvidal.cse.sc.edu/library/jade.pdf> (accessed on 14 April 2020).
35. Melton, M. Alexa Skill Voice Interface for the Moodle Learning Management System. *J. Comput. Sci. Coll.* **2019**, *35*, 26–35.
36. Stoyanov, S.; Popchev, I.; Doychev, E.; Mitev, D.; Valkanov, V.; Stoyanova-Doycheva, A.; Valkanova, V.; Minov, I. DeLC Educational Portal. *Cybern. Inf. Technol.* **2010**, *10*, 49–69.
37. Amazon. Alexa S Kill K it D Ocumentation, 2020. Available online: <https://developer.amazon.com/en-US/docs/alexa/alexa-skills-kit-sdk-for-java/handle-requests.html> (accessed on 17 April 2020).
38. Moodle. Mobile web services—MoodleDocs, 2020. Available online: <https://docs.moodle.org> (accessed on 16 April 2020).
39. Perez, M. Method and System of USING OAuth2 to secure Neighbor Discovery. U.S. Patent 10,248,365, 30 December 2016.
40. Holden, T. Making Your Devices Speak. Integration between Amazon Alexa and the Managed IoT Cloud. Master's Thesis, UiT Norges Arktiske Universitet, Tromsø, Norway, 2018.
41. Jones, M.; Bradley, J.; Sakimura, N. JSON Web Token (JWT). RFC 7519, 2015. Available online: <https://tools.ietf.org/html/rfc7519> (accessed on 18 April 2020). [CrossRef]
42. Agrawal, M.; Mishra, P. A Comparative Survey on Symmetric Key Encryption Techniques. *Int. J. Comput. Sci. Eng.* **2012**, *4*, 877.
43. Usman, M.; Shoukat, I.A.; Malik, M.S.A.; Abid, M.; ul Hasan, M.M.; Khalid, Z. A Comprehensive Comparison of Symmetric Cryptographic Algorithms by Using Multiple Types of Parameters. *Int. J. Comput. Sci. Netw. Secur.* **2018**, *18*, 131–137.

44. Lucks, S.; Callas, J. The Skein Hash Function Family. 2009. Available online: <http://www.skein-hash.info/sites/default/files/skein1.3.pdf> (accessed on 19 April 2020).
45. Mesran, M.; Abdullah, D.; Hartama, D.; Roslina, R.; Asri, A.; Rahim, R.; Ahmar, A.S. Combination Base64 and Hashing Variable Length for Securing Data. *J. Phys. Conf. Ser.* **2018**, *1028*, 012056. [[CrossRef](#)]
46. Brooke, J. SUS: A retrospective. *J. Usability Stud.* **2013**, *8*, 29–40.
47. Laugwitz, B.; Held, T.; Schrepp, M. Construction and Evaluation of a User Experience Questionnaire. In *HCI and Usability for Education and Work. USAB 2008*; Holzinger, A., Ed.; Springer: Berlin/Heidelberg, Germany, 2008; Volume 5298, pp. 63–76.
48. Saiz-Manzanares, M.; Marticorena-Sánchez, R.; Ochoa-Orihuel, J. *UBU Voice Assistant Computer Application Software*; BU-69-20; General Registry of Intellectual Property: Madrid, Spain, 2020.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).