ARIA: Efficient Music Festival Manager

David A. Mar, Jr., Emily Huang, Vladislav B. Savranschi
Renee T. Iinuma, Wesley Kepke, Ernest Landrito, Kyle Lee
Cynthia R. Harris, Sergiu M. Dascalu, Frederick C. Harris, Jr.

Department of Computer Science and Engineering
University of Nevada, Reno
Reno, NV 89557 USA
dmar@nevada.unr.edu

Abstract

Administration, Registration, and Information Assistant (ARIA) is a plugin that runs on the WordPress platform. The purpose of ARIA is to manage event creation, student and teacher registration, event scheduling, and document generation for the Reno Youth Music Festival, an event that is run by the Northern Nevada Music Teachers Association (NNMTA). In the past, NNMTA managed the registrations and payments for these events through pen-and-paper methods. This can easily lead to many hours of tedious work and can lead to human errors in the process. ARIA is a proposed solution to improve the efficiency, reliability, and security of the event management process.

keywords: event management, festival, music, NNMTA, registration, WordPress

1 Introduction

Administration, Registration, and Information Assistant (ARIA) is a WordPress plugin which has been under development for the past two years by the authors and has been sponsored by the Northern Nevada Music Teachers Association (NNMTA), a music organization in Reno, Nevada. The software assists in running the Reno Youth Music Festival.

In this popular event, music students of various levels perform in front of judges and, afterwards, receive feedback and ratings from them. The music festival occurs twice every spring: one for the upper level division and one for the lower level division.

In the past, running this event required paperwork to register students and teachers, schedule the students, and generate the necessary documents (e.g. performance times, volunteer roles) for the day of the event. Processing the data by hand can be tedious, time-consuming, and prone to error.

By utilizing ARIA, the organization can more efficiently set up the music festival by letting the plugin manage the forms and data processing. Furthermore, students and teachers can quickly register for the event and monitor their performance status.

The rest of this paper is structured as follows: Section 2 goes over music organizations and a more detailed description of the music festival that uses ARIA. Section 3 discusses the literature of online event managers. Section 4 goes over the requirements and use cases of the system. Section 5 describes the resulting program, and Section 6 discusses future works. Section 7 concludes the paper with a summary.

2 Background

Music organizations work to support music in endeavors ranging from business to the science of music. One organization, the Music Teachers National Association (MTNA), was established in 1876 to support music education. This national group holds conferences, training, competition, and festivals, all to promote music literature, provide opportunities for young musicians develop their skills, and nurture the professional growth of music teachers [4]. The MTNA is affiliated with many regional music organizations, including the Nevada Music Teachers Association (NMTA) [5]. Within the NMTA is a local organization called the Northern Nevada Music Teachers Association (NNMTA).

NNMTA has been around since 1973 and became an affiliate with MNTA in 1978. It was formed in Reno with the goal to provide opportunities in music education and performance for both music students and teachers. On their website, resources on effective teaching as well as links to state and national conferences are provided. For students, NNMTA offers festivals and competitions, one of them being the Reno Youth Music Festival [6].

The following describes this music festival in greater detail. At the beginning of the event, a registration period opens for students to enter basic information and their level. For each entry submitted by a student, that student's teacher must fill out the students' pieces, the estimated performance time, and other information. Once the registration period ends, the event organizer or chairman creates and publishes a schedule, listing performance

information for students and assignments roles for judges and volunteers.

There are three types of sessions students can enter: traditional, non-competitive, and master class. In a traditional session, the students perform their pieces and receive feedback and ratings from the judges. The possible ratings are, starting with the lowest, “Needs Attention”, “Excellent”, “Superior”, and “Superior with Distinction”. If a student receives a “Superior” or a “Superior with Distinction” rating, they are asked to perform again in a recital called the Command Performance. This recital usually happens the week after the music festival. Non-competitive sessions are like traditional sessions except that the students do not receive a rating.

Master classes differ from traditional and non-competitive sessions as the judge not only rates and provides feedback to the students but also gives a mini lesson on the pieces performed. Master class judges are usually world-renowned musicians and provide valuable information on technique, practicing, and performance; for these, it is common to see music teachers and students attending these events. In master classes, students receive ratings of “Master Class” or “Command Performance”. Those with “Command Performance” ratings may perform again at the Command Performance.

3 Literature

Numerous event and registration management systems can be found online to create forms and process data. Each offers unique functionalities to allow them an edge in the market. One service, Eventbrite, lets users set up events, advertise on social media, and sell tickets. Its website and mobile application also provide a way to explore and find events in the local region. On the developer side, it provides tools to be integrated with WordPress [1].

Another service, Whova, also provides an interface to set up events and programs. Moreover, users are given the ability to connect and network with companies, presenters, and other attendees through this application [9].

Other registration management systems are available for more general uses. For example, RegPack advertises itself as an online registration application that can accept online payments, manage users, send emails, and perform various CMS-like automations [8]. Another product, RegFox, covers similar features and can be integrated with WordPress [7].

While these products can speed up the registration process, they do not have additional features needed for the Reno Youth Music Festival. Such features include scheduling performers based on their levels and preferences and generating documents based on the schedule. By developing ARIA to support registration, scheduling, and document creation, it can speed up data processing throughout the music event.

ARIA also runs as a music-based application, which leads to some unique aspects that are not commonly seen in the market. This software allows the user to upload a syllabus listing the pieces that students can then select from their level. Additionally, its scheduler takes in consideration student performance times and the number of times a piece was selected. Along with making the Reno Youth Music Festival easier to run and manage, further explorations into these features may provide insight as to how they can be generalized and used for similar music events.

4 Implementation

Meetings were held with Mrs. Cynthia Harris, Vice President of NNMTA, and Dr. Frederick Harris, the technical advisor for ARIA, to discuss the software requirements. These are essential for a couple reasons. First, the meeting helps solidify the requirements desired by NNMTA. At the same time, preliminary ambiguities are removed, reducing the amount of lost development time to rewrite software. Some examples of the specifications are the required fields for registration forms, features accessible only to the chairman, requirements for scheduling, and the types of documents needed.

During these meetings, the structure of the system is laid out. Duplicated features are translated into generalizable and reusable functions. The determined platform and database manager lets developers know what tools are available. Such information that is deduced in the meeting usually results in a well-organized and maintainable system.

Finally, the meeting establishes deadlines for software development and feasibility in the given timeframe. Thus, requirements must be organized and prioritized to ensure that the most important features are ready on time. These set goals are also used to measure progress and indicate if any plans need to be altered.

The primary specifications of ARIA are described in the following sub-sections. In particular, these define the functional requirements, the non-functional requirements, and the use cases that must be met by the software. Description of each can be found in Ian Sommerville’s *Software Engineering* [3].

4.1 Functional Requirements

Functional requirements describe system roles, components, and how they interact. Listed below are the main functional requirements for ARIA:

1. The system shall provide a private web portal for administrators to operate the event.
2. The system shall provide the forms required to run the music festival. The major forms are those used for

creating a festival event, registering students and teachers, and scheduling the students.

3. The system shall use a database to store and retrieve information.
4. The system shall provide a function to upload a music syllabus containing a list of pieces allowed in each level.
5. The system shall provide an option to add, modify, and delete teacher and student information as the event progresses.
6. The system shall provide a payment transaction system, essential to completing the student registration process.
7. The system shall provide a web portal for teachers where they can view a list of their students, their contact information, and other relevant data.
8. The system shall provide a scheduling feature to generate an event schedule based on provided input.
9. The system shall provide an interface to modify the schedule. The user should be able to save or cancel any changes made.
10. The system shall have a document generator to produce the documents necessary for running the event.
11. The system shall provide a way for judges to input scores for student performances.
12. The system shall provide a method to publish a schedule to teachers and parents via email.

4.2 Non-functional Requirements

The following requirements describe constraints that the system must abide by as its features are being used.

1. The web pages should be viewable and functional on all current modern web browsers (e.g. Chrome, Firefox, Edge, Safari).
2. The system is to be implemented in PHP, HTML, CSS, and JavaScript.
3. The system shall operate on the WordPress platform as an extensible plugin that can be activated and deactivated [10].
4. The system shall utilize the Gravity Forms WordPress plugin to handle form generation, database records, and payment transactions [2].

4.3 Detailed Use Cases

The following describes the interaction between users and the system as seen in Figure 1.

1. **Access Chairman Portal**
   This will provide the chairman controls for the event. Some controls include viewing student and teacher information, generating schedules and documents, and modifying the information.
2. **Upload Music File**
   The chairman can upload a list of pieces that can be played in the event. The list to be uploaded must follow a specific format; otherwise, an error will occur.
3. **Create Event**
   The chairman can create an event after providing information about that event, such as event name, dates, location, and fees. The necessary registration forms and databases will be created for the event.

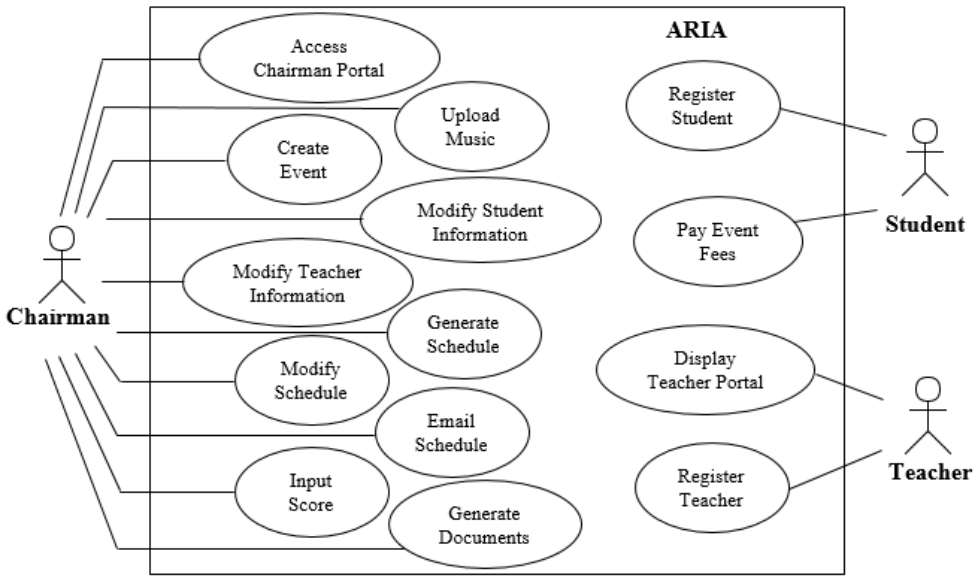

Figure 1: Use case diagram of ARIA. System functionalities are paired with users of the system.

4. **Register Student**
   After the student has filled out the student registration form, the data will be collected and stored in the student database.
5. **Pay Event Fees**
   The student will pay for the event after they have filled out their form. This must be done to complete the student portion of the registration. The total cost will depend on the student level.
6. **Display Teacher Portal**
   This will create a list of students that have completed their portion of the registration process. Teachers here can view information on their students, such as registration status and performance times.
7. **Register Teacher**
   The teacher provides additional information regarding their student. One entry must be completed for each student.
8. **Modify Student Information**
   The chairman can change any information regarding a student that is registered for a given event.
9. **Modify Teacher Information**
   The chairman can change any information regarding a teacher that has students or is judging in a given event.
10. **Generate Event Schedule**
    This provides a basic structure for the chairman to generate a schedule for an event. Once the type of event and event options have been determined, it will collect the necessary data and generate a schedule.
11. **Modify Event Schedule**
    Once a schedule has been created, the chairman can make any changes to the schedule. The chairman will have the option to either save or discard the changes.
12. **Generate Documents**
    The chairman can create documents based on the schedule information, the students, and the teachers.
13. **Input Judge Score**
    This will be done after the student has performed and the judges have written their score. The chairman will input scores to a form. Once completed, data from the forms will be extracted and stored in the database.
14. **Email Schedule Information**
    This allows the chairman to email parents and teachers regarding information on their child or student's scheduled performance.

5 Results

Figure 2 shows a context diagram displaying the primary systems of ARIA required to fulfill the system requirements. The following subsections provide an overview of portals, registration, scheduling, and document creation.

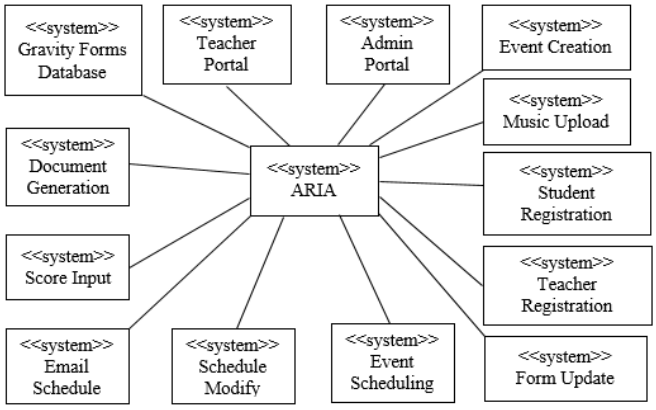

Figure 2: Context diagram of ARIA. All the system components are used to meet the requirements specified by NNMTA.

5.1 Portals

The portals are the primary user interface for event management and administration. The Admin Portal, seen in Figure 3, is accessible only by an administrator-level account and provides navigation for the user to create and schedule events, edit student and teacher information, and generate documents. This portal also includes the Event Viewer which gives the administrator an at-a-glance view of all registered students, teachers, and schedules.

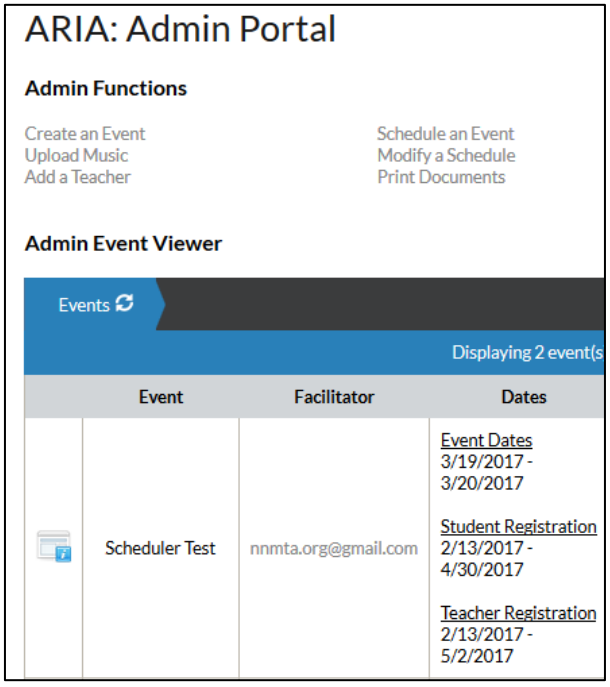

Figure 3: An example of a portal. This administrator portal provides access to all ARIA functionalities and allows the user to view, modify, add, and delete data related to the event.

Teachers that participate in the event are also provided access to their own Teacher Portals. From here, a teacher can view and manage their students' registrations. They can also view their students' scheduled performance times and any volunteer tasks assigned to them by the event facilitator.

5.2 Registration

The registration component consists of student and teacher registration pages. The student registration form can be seen in Figure 4. Parents fill out this form, providing student information along with preferred performance times. Once the form is submitted, they then pay the fees through PayPal, an application supported by Gravity Forms. The registration must be completed by the date the chairman specifies.

Student performance pieces are assigned by their teacher during teacher registration. Further information, such as event category, student theory score, and teacher volunteer preferences, are also collected during teacher registration.

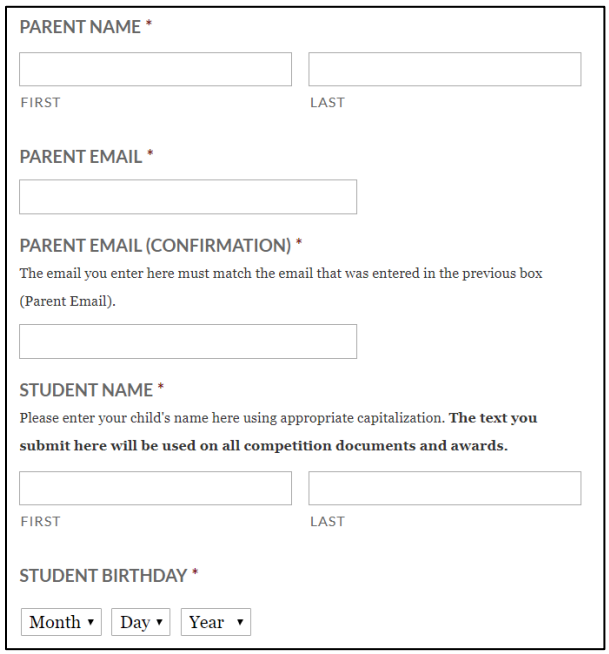

Figure 4: A section of the student registration form. This and a variety of other forms will prompt fields necessary for running the music festival.

5.3 Scheduling

The scheduler assigns students, judges, and volunteers to sections where they will perform their roles. When creating the schedule, the event organizer enters in necessary information, such as the number of time blocks and sections, the start times, and the duration of a section. If students cannot be accommodated within the given parameters, the organizer will receive an error message. The scheduler can be used for both the music festival and the Command Performance.

Once the schedule is created and displayed, as shown in Figure 5, the chairman can use a schedule interface to adjust section information, student section assignment, and performance order. Students can also be moved to different days and time blocks.

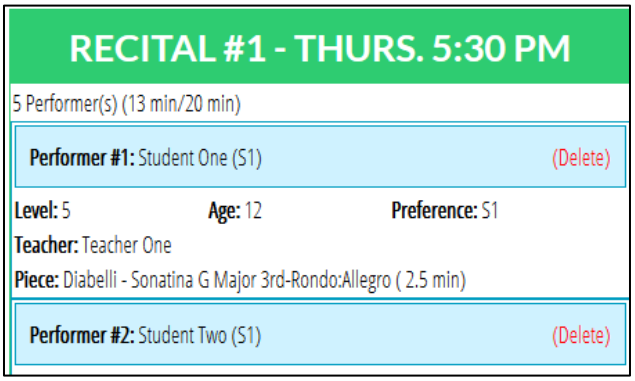

Figure 5: A snapshot of the Command Performance schedule. Information on the session and students are displayed. The event organizer can reorder students or move them to different sessions.

5.4 Document Generator

A wide variety of documents must be created and printed for the music festival. One example can be seen in Figure 6. These documents provide information on student performance times and locations as well as tasks assigned by volunteers and judges. The administrator can also select which documents to download by navigating to the Document Generation page.

Documents can be generated in RTF and CSV formats for traditional, non-competitive, and master class sections. They can also be used to generate documents for the Command Performance.

	A	B	C	D	E	F	G
1	Scheduler Test						
2	Section	Format	Teacher	Student	Age	Level	Piece 1
3	Saturday,	Traditiona	Teacher 1	Student A	8	1	Minuet, F
4		Traditiona	Teacher 1	Student B	10	1	Folk Song
5		Traditiona	Teacher 2	Student C	10	1	Distant Be
6		Traditiona	Teacher 2	Student D	11	1	Etude, D r
7							
8	Saturday,	Traditiona	Teacher 3	Student G	9	1	The Cucko
9		Traditiona	Teacher 4	Student H	9	1	Minuet, F

Figure 6: An example document generated using data from the scheduler and registration forms. Documents are generated throughout various stages of the music event.

6 Discussion and Future Work

In terms of core functionality, ARIA has been fully implemented and can be used to support NNMTA and the Reno Youth Music Festival. It has been used a total of four times for the event: twice for the 2017 upper and lower festivals, and twice for the 2016 upper and lower festivals. Teachers and parents of students were satisfied with the fast and easy registration process, while the chairman was pleased with the improved efficiency of running the music festival.

With the base completed, additional adjustments on ARIA can be made to meet the chairman's requests and make the plugin more user friendly.

ARIA is currently designed to be used solely for the NNMTA music festivals. One direction for continued development of the application would be to extend its usability beyond its current scope. To do so, it may be best to make the system independent of the WordPress platform and the Gravity Forms plugin. While these have provided functionalities to speed up development time, these platforms have also provided constraints. Additionally, removing Gravity Forms would reduce the cost to maintain ARIA, while removing the WordPress dependency would expand the potential audience.

As this application was being developed, it was found that features had to be modified to improve the usability of the plugin. This emphasizes the importance of making the system as versatile as possible so that maintenance and further development can be done with minimum effort.

7 Conclusion

ARIA is an event manager that assists NNMTA in running the Reno Youth Music Festival. It handles and processes the paperwork done in registration, scheduling, and document generation. As a result, running the music festival has become more efficient and less tedious and error prone. It has been fully implemented as a WordPress plugin, but developing the system independently from the platform would increase its flexibility and accessibility. In its current state, ARIA has become a reliable application with many potential areas to expand and improve.

Developing a general event manager for music festivals and similar events can prove to be a challenge as each organization has its own specific requirements. Features such as scheduling and document generation can have many variabilities, making it incredibly difficult if not impossible to reuse a system without making major modifications. ARIA was specifically developed to support a particular event, but it stands as a stepping stone towards what can be done to create a general event manager for music events.

Acknowledgement

This material is based in part upon work supported by the National Science Foundation under grant numbers IIA-1329469 and IIA-1301726. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Eventbrite. Eventbrite – Discover Great Events or Create Your Own & Sell Tickets. https://www.eventbrite .com/, n.d. Last Accessed: June 27, 2017.

[2] Gravity Forms. Gravity Forms is the Easiest Tool to Create Advanced Forms for Your WordPress Powered Website. http://www.gravityforms.com/, n.d. Last Accessed: June 27. 2017.

[3] Ian Sommerville. *Software Engineering*. Addison-Wesley, Harlow, England, 9 edition, 2011.

[4] MTNA. Working for a more musical tomorrow. http://www.mtna.org/about-mtna/, n.d. Last Accessed: June 27, 2017.

[5] NMTA. About the NMTA. http://www.nvmta.com/about.php, n.d. Last Accessed: June 27, 2017.

[6] Northern Nevada Music Teachers Association. Mission Statement. http://www.nnmta.org/home-2/home/, n.d. Last Accessed: June 27, 2017.

[7] RegFox. RegFox Event Management Software. https://www.regfox.com/, n.d. Last Accessed: June 27, 2017.

[8] Regpack. Online Registration Software with Regpack. https://www.regpacks.com/, n.d. Last Accessed: June 27, 2017.

[9] Whova. Event App and All-in-One Event Management Software. https://whova.com/, n.d. Last Accessed: June 27, 2017.

[10] WordPress.org. Meet WordPress. https://wordpress .com/, n.d. Last Accessed: June 27, 2017.