



OBJECT ORIENTED PROGRAMMING

FRÉDERICK SANTIAGO TIPÁN MORENO

GABRIEL NICOLAS VIVANCO RAZA

JEFFERSON DAVID YEPEZ MORAN

NRC: 14575

SANGOLQUÍ 2023



1.- INTRODUCTION	2
1.1.- Purpose	2
1.2.- System Scope	2
1.3.- Definitions, Acronyms and Abbreviations	2
1.4.- References	3
1.5.- Document Overview	4
2.- GENERAL DESCRIPTION	4
2.1.- Product Perspective	4
2.2.- Product Features	4
2.3.- User Characteristics	4
2.4.- Restrictions	4
2.5.- Assumptions and Dependencies	5
2.6.- Future Requirements	5
2.7.- Class Diagram	5
3.- SPECIFIC REQUIREMENTS	5
3.1.- Interfaces	5
3.2.- Functional requirements	5
3.3.- Non-functional requirements	6

1.- INTRODUCTION

This document provides readers with information about the Organivent software project; being a system developed specifically for the organization of artistic events for the company Feel the Tickets. The document is following the format given by IEEE 830.

1.1.- Purpose

Present the main functionalities of a software for the organization of large-scale events in the city of Quito, based on specific requirements so that they allow providing an effective summary of the resources used in each presentation.

1.2.- System Scope

The system to be developed is called “Organivent”, being a system that will have the capacity to store the most important information when an event takes place. However, the domain of this project is for artistic events, since it will work with part of the information provided by the company Feel the Tickets, since due to confidentiality issues it is not possible to have full access to 100% of the investment of each event.

For this reason, Organivent will take into account relevant aspects such as the place, the capacity of the place, the artist, the equipment and the staff involved along with their respective schedules to generate an approximation of the expenses to carry out an event. In this way, each company administrator will be able to view the information to better distribute their resources for future presentations.

1.3.- Definitions, Acronyms and Abbreviations

- **Artist:** A person who practices any of the various creative arts, in this case they will be focused on musical performances.
- **Sponsor:** An individual or organization that pays some or all of the costs involved in staging a sporting or artistic event in return for advertising.

- **Staff:** All the people employed by a particular organization from different work areas.
- **Schedule:** A list of the time involved for a certain event, this includes a range for the date and hour when it takes place.
- **Event:** Anything that happens, especially something important or unusual.
- **Tickets:** A small piece of paper or card given to someone, usually to show that they have paid for an event, journey, or activity.
- **Administrator:** Someone whose job is to control the operation of a business, organization, or plan.
- **Java:** Programming language designed for the object oriented paradigm.
- **IEEE 830:** Standard covering software requirements.

1.4.- References

Title	Rute	Author
IEEE Guide for Software Requirements Specifications	https://ieeexplore.ieee.org/document/278253/references#references	IEEE
DESARROLLO DE UN SISTEMA WEB PARA ADMINISTRACIÓN DE EVENTOS Y CONTROL DE ENTRADA Y SALIDA DE EMPLEADOS	http://repositorio.puce.edu.ec/bitstream/handle/22000/8105/Tesis-GWMM-Entrega-vFinal.pdf?sequence=1&isAllowed=y	GUSTAVO WLADIMIR MOYANO MEJÍA
What is an Event Management System?	https://www.emssoftware.com/resources/blog-posts/what-event-management-system#:~:text=An%20event%20management%20system%20allows.planning%2C%20reporting%2C%20and%20more.	Accruent EMS
What is Event Management	https://eventscase.com/blog/wh	Mentxu Sendino

Software?	at-is-event-management-software	
How to Organize a Concert Event: Checklists and Guidebook	https://www.viptogo.com/blog/how-to-organize-a-concert-event	Viptogo

1.5.- Document Overview

The document is made up of three parts, the first (already displayed) broadly shows the main purpose of the software to be developed. Later, both the user and the client will be mentioned in more detail, as well as the main characteristics of the system. Dependencies and restrictions that will pose requirements to be met in the future will also be mentioned. Finally, both the functional and non-functional requirements will be detailed, which are determined based on the needs requested by the customer to provide a quality system.

2.- GENERAL DESCRIPTION

2.1.- Product Perspective

The system will be focused on managing and organizing everything that will be needed on the staff side, the personnel, the equipment, the budget, the capacity of the place and collect all the information to obtain approximate costs to hold the event. This system will not depend on other programs.

2.2.- Product Features

- Creation and editing of events.
- Management of artists and participants.
- Scheduling of activities and timetables.
- Calculate an approximate cost.

2.3.- User Characteristics

The system will be used by the manager and administrators.

Manager: Claudia Hidrobo
Education level: Superior
Experience: Long time doing events

Administrator: Miguel Vivanco

Education level: Superior

Experience: Long/medium time doing events

The system does not require experience or extensive knowledge in systems, thanks to the system's design.

2.4.- Restrictions

- Compatibility with specific web browsers.
- Not full access to the investment for each event.

2.5.- Assumptions and Dependencies

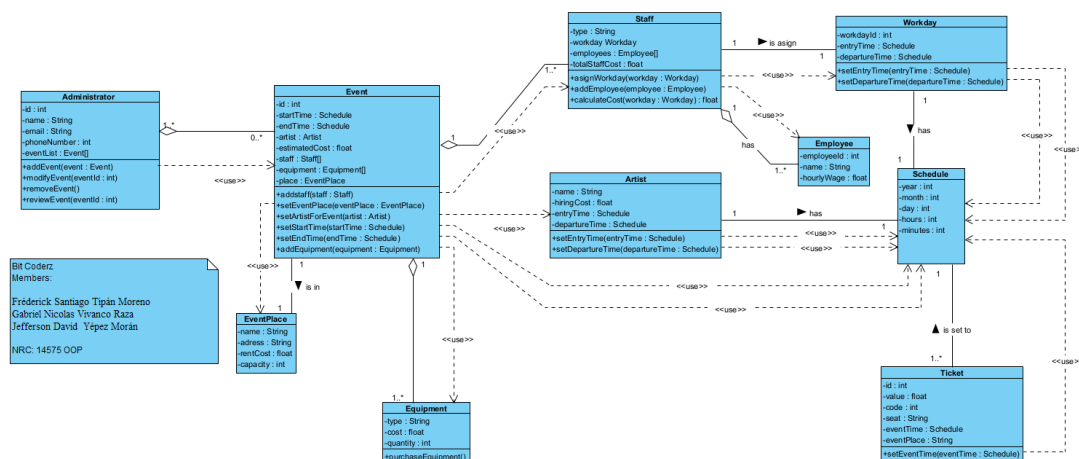
It is assumed that the main requirements will not suffer a major change in the future, unless the company decides to offer another service during the events that may add another aspect that could increase the investment cost. It may also suffer changes if it decided that the system would need a visual interface and a better database system.

2.6.- Future Requirements

The main future requirement would be to implement a more attractive interface for the user, trying to implement buttons to confirm actions when using the software and not lean on menus too often.

Also, the system could implement a built-in function to generate an analysis based on all the data stored for each event, that way the administrator will have an easy way to visualize the information.

2.7.- Class Diagram



3.- SPECIFIC REQUIREMENTS

3.1.- Interfaces

Due to the object oriented programming course we will be using java, the interface will be used from the console. That doesn't mean that the interface is going to be simple, because it is going to have well-distributed information., using an easy to use menu to navigate throughout the system.

3.2.- Functional requirements

- **Participant Registration:**

The system should allow users to register for the event, providing necessary information and enabling the selection of options such as ticket type, additional workshops, etc.

- **Event Planning:**

It should be possible to create and plan events, allocate resources, define schedules, and manage the scheduling of activities and performances.

- **Exhibitors and Artists Management:**

It should facilitate the management of exhibitor and artist profiles, including the allocation of exhibition spaces and logistical coordination.

- **Evaluation and Reviews System:**

The software should facilitate the creation of evaluation and review systems for artists, activities, and the event as a whole, allowing attendees to provide feedback.

- **Real-Time Data Analysis Tools:**

It should offer real-time data analysis tools to assess event performance, attendee engagement, and other key indicators.

3.3.- Non-functional requirements

- **Response Time:**

It must ensure an acceptable response time for users, ensuring that all the operations don't have a huge delay.

- **Usability and User Experience:**

The software should have an easy-to-use, intuitive, and accessible interface, ensuring a positive user experience.

- **Availability:**

It must guarantee high availability, minimizing planned and unplanned downtime to ensure that the system is available when needed.

- **Adaptability and Customization:**

It should be adaptable to different types of events and allow customization to meet the specific needs of each event organizer.

- **Maintainability:**

The software should be easy to maintain and update to incorporate new features, correct errors, and adapt to changes in event requirements.

- **Fault Tolerance:**

It should be fault-tolerant, capable of recovering from errors, and minimizing the impact on the user experience in case of problems.