



GETTICK

REQUIREMENT ANALYSIS PLAN

051900000081 - Safiye Defne Kargı
052000000034 - Muhammet Şancı
051900000104 - Hazal Karataş
052000000761 - Berat Duran

INTRODUCTION

Gettick, new event management system is a comprehensive platform designed to streamline the planning, organization, and ticketing of events. This system will provide event organizers and attendees with a seamless experience, offering a wide range of features to enhance event management.

In this requirement analysis plan; there are viewpoints and their descriptions, requirements and their definitions, classification of the requirements, requirement tracibility matrix, main scenario use cases, and the domain model.

TABLE OF CONTENTS

COVER	1
INTRODUCTION	2
PRINCIPILE VIEWPOINTS	3
VIEWPOINT DIAGRAM	4
VIEWPOINTS LIST	5
REQUIREMENTS	6
PRIORITIZATION	9
PRIORITY TABLE	10
TRACIBILITY MATRIX	11
MAIN USE CASES	13
DOMAIN MODEL	16
CONCLUSION	17

PRINCIPLE VIEWPOINTS

These are the types of Gettick's principle viewpoints that we will discuss in detail in the following pages:

Functional Viewpoint

This focuses on the functionality or features of the system. It describes what the system does in terms of its operations, services, and capabilities.

Structural Viewpoint

This addresses the organization and composition of the system's components. It includes information about the system's architecture, modules, components, and their relationships.

Behavioral Viewpoint

This viewpoint emphasizes how the system behaves or responds to different inputs and stimuli. It includes details about state transitions, events, and interactions.

Informational Viewpoint

This deals with the data and information managed by the system. It includes data models, databases, data flows, and storage mechanisms.

Development Viewpoint

This pertains to the organization of the system during its development process. It may include information about code repositories, build processes, development tools, and environments.

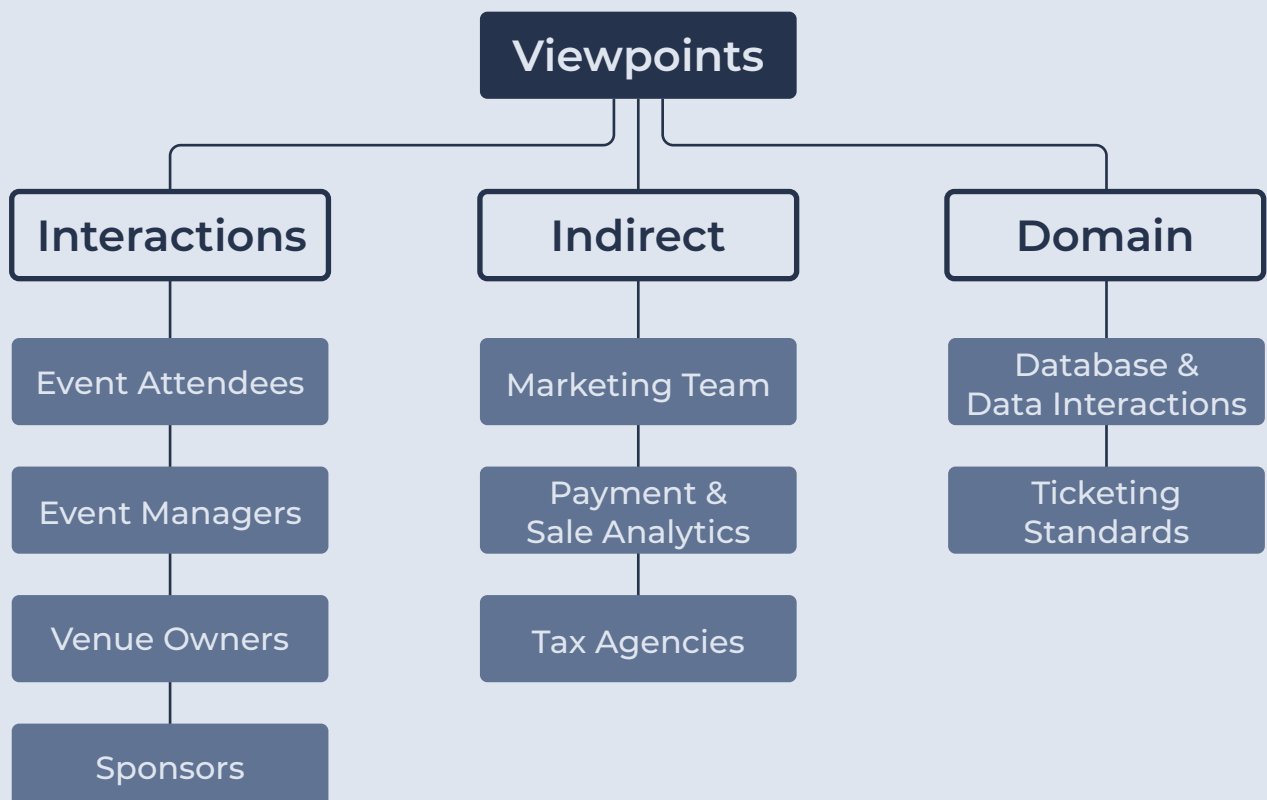
PRINCIPLE VIEWPOINTS

First, we will specify the stakeholders of this project and the hierarchical viewpoint diagram.

GETTICK STAKEHOLDERS

- EVENT ATTENDEES - main user
- EVENT ORGANIZERS - main user
- PAYMENT GATEWAY PROVIDERS
- VENUE OWNERS
- DEVELOPERS
- MARKETING TEAM
- SPONSORS & PARTNERS
- TAX AGENCIES

VIEWPOINT DIAGRAM



VIEWPOINT LIST

In this table, we provided descriptions of the viewpoints and related requirements.

Viewpoint and Description		Relation
Event Attendees	Functional	R1 R2 R3 R4 R5
This includes functionalities for tracking, collecting, and managing attendee registration; providing information for ticketing.		
Event Managers	Functional	R6 R7 R8
This viewpoint focuses on functionalities related to setting up and customizing events, including details, date, time and venue.		
Venue Owners	Functional & Structural	R9 R10 R11
This viewpoint includes functionalities related to onsite check-in, registration, and seating chart design.		
Sponsors	Functional	R12 R13
Functionalities related to sponsoring/partnering events and/or platform.		
Marketing Team	Functional	R14 R15
This functionalities includes marketing and event campaign activities.		
Payment & Sale Analytics	Functional & Behavioral	R16 R17 R18 R19
This viewpoint includes providing payment gateways for ticket sale and managing the analytics for sales.		
Tax Agencies	Informational	R20
This viewpoint is related to tax payment system and providing reports for taxes to both government and main stakeholders.		
Database & Data Interactions	Informational & Development	R21 R22
This viewpoint is related to designing and maintaining database systems for platform and providing data and analytics for processes like CRM.		
Ticketing Standards	Functional & Behavioral	R23 R24
This covers functionalities for online ticket sales, registration, real-time availability checks, providing QR codes, virtual seating previews.		

REQUIREMENTS

In this table, we provided requirements for Gettick and also classifications for them, depending on function and lifetime.

Requirement and Description		Class
R1	Account Management A user can create account as event attendee, can delete the account and set interests for the account.	Functional Enduring
R2	Event Navigation An event attendee can search and filter events by city, date, time and ticket price criterias.	Functional Enduring
R3	Order Management An event attendee can buy tickets for events, can cancel purchased tickets, display QR code for purchased tickets, and view registrated events.	Functional Enduring
R4	Ticket Placing An event attendee can make payment and buy tickets by using different kind of payment types.	Functional Enduring
R5	Event Reminder An event attendee can create notifications for upcoming events for which they are registered.	Functional Enduring
R6	Event Management The event organizer can create and delete events, manage date and time, prices, and can send notifications those who registered.	Functional Enduring
R7	Attendee Management The event manager can view the attendee list and track check-in.	Functional Enduring
R8	Check Reports The event manager can manage and track sales reports, demographic reports, and other types of reports.	Functional Enduring
R9	Venue Management Venue owners can give users the exact address of the event venue and provide a seating plan to determine seating arrangements.	Functional Enduring

REQUIREMENTS cont.

Requirement and Description		Class
R10	Real Time Report	Functional
Venue owners should be able to access data such as seating availability and track commission sales and usage status of the venue by time.		Enduring
R11	Check-In	Functional
Venue owners can offer users the opportunity to check-in quickly and securely on site via QR codes provided in the platform.		Enduring
R12	Event Sponsoring	Functional
A sponsor has the authority to provide support for events to be utilized in the platform.		Enduring
R13	Sponsor Manager Communication	Functional
A sponsor can communicate with the event manager via platform.		Enduring
R14	Campaign and Ad Creation	Functional
The marketing team has the capability to create advertisements for the events in the platform.		Enduring
R15	Campaign Related Reports	Functional
The marketing team is able to review reports on campaign-related sales and return on investment.		Enduring
R16	Payment System	Functional
The third party payment service integrated for payments should provide different payment systems.		Enduring
R17	Statistic Reports	Functional
The system should provide a dashboard for the event organizers or managers to view and analyze the sales report and statistics, such as the number of tickets sold, the average ticket price, etc.		Enduring
R18	E-Receipt	Functional
The third party payment service integrated for payments should send electronic receipts to users.		Enduring

REQUIREMENTS cont.

Requirement and Description		Class
R19	Provide Refund	Functional
The third party payment service ensures user satisfaction through efficient refund processing.		Enduring
R20	Tax Calculation	Functional
Tax calculation algorithms included in the payment system should calculate taxes and inform users while making payment.		Enduring
R21	Database Management	Domain
The database is designed with appropriate tables, fields, keys, indexes, and constraints to ensure data integrity, security, and performance.		Enduring
R22	Customer Relationship Management	Functional
The system should provide a CRM to the platform for helping the event organizers or managers to improve customer satisfaction, retention, and as well as to increase sales and revenue.		Enduring
R23	Ticket Standards	Non-Func
The ticketing system must generate and assign unique, alphanumeric REF numbers to each purchased ticket, and QR code should involve it.		Enduring
R24	Real Time Synch	Non-Func
The system synchronizes the data in real time, to ensure that the information is consistent and up-to-date across all channels.		Enduring
R25	Regulatory Compliance	Non-Func
The system must comply with relevant data protection regulations, such as KVKK (For TR), and industry standards for payment processing.		Enduring
R26	User Friendly UI/UX	Non-Func
The system should provide a user-friendly interface that allows both event organizers and attendees to navigate and use the platform easily and must be compatible with all major smartphone and tablet devices.		Volatile
R27	Launch Time	Non-Func
On average, the app must open within 1 seconds, but it must not take longer than 2 second to open under any circumstances.		Volatile

PRIORITIZATION

We measured every requirement by benefit, risk, dependency and time sensitivity and wrote down average scores and priorities.

Benefit: This is the benefit that the platform will gain if a specific need is met first and foremost.

- 1 Loosely Beneficial
- 5 Highly Beneficial

Risk: The risk is whether or not the need will yield the expected value and meet the requirement.

- 1 No Risk If It Doesn't Meet
- 5 High Risk If It Doesn't Meet

Dependency: This expresses the dependency of other requirements on the current requirement

- 1 No Dependency On It
- 5 High Dependency On It

Time Sensitivity: It indicates how early it should be implemented, that is, urgency of the requirement.

- 1 Not Urgent
- 5 Highly Urgent

We then calculated average importance for every requirement, and declared its priority.

- 0.00 – 2.00 Low Priority
- 2.00 – 3.75 Medium Priority
- 3.75 – 5.00 High Priority

Check next page for overall priority table.

PRIORITY TABLE

ID	Benefit	Risk	Dependency	Time Sensitivity	Average	Priority
R1	5	5	5	5	5.00	H
R2	4	3	1	3	2.75	M
R3	5	5	5	5	5.00	H
R4	5	5	5	5	5.00	H
R5	2	1	1	1	1.25	L
R6	5	5	5	4	4.75	H
R7	3	3	2	3	2.75	M
R8	5	1	3	3	3.00	M
R9	4	3	2	4	3.25	M
R10	5	4	3	5	4.25	H
R11	5	5	5	5	5.00	H
R12	5	1	2	2	2.50	M
R13	3	3	1	4	2.75	M
R14	5	5	1	5	4.00	H
R15	5	5	1	5	3.50	M
R16	5	1	3	5	3.50	M
R17	4	3	5	3	3.75	H
R18	5	2	4	3	3.50	H
R19	2	3	5	1	2.75	M
R20	3	4	1	3	2.75	M
R21	3	5	4	6	4.50	H
R22	5	2	2	4	3.25	M
R23	4	1	2	4	2.75	M
R24	5	4	3	5	4.25	H
R25	5	5	5	5	5.00	H
R26	5	3	2	3	3.25	M
R27	5	3	1	3	3.00	M

TRACIBILITY MATRIX unfor.

ID	Source Tracibility	Requirement Tracibility	Design Tracibility
R1	Event Attendee	Independent	User Accounts
R2	Event Attendee	R6 - R10 - R23	Event Structure
R3	Event Attendee	R4 - R7 - R8 - R10 - R18	Ticketing
R4	Event Attendee	R3 - R16 - R18 - R19 - R23	Ticketing
R5	Event Attendee	R6	Notification System
R6	Event Manager	R2 - R5	Event Structure
R7	Event Manager	R3 - R8	Reporting
R8	Event Manager	R3 - R7 - R17	Reporting
R9	Venue Owner	Independent	Venue Structure
R10	Venue Owner	R3	Reporting
R11	Venue Owner	Independent	Ticketing
R12	Sponsor / Partner	R13	Sponsoring
R13	Sponsor / Partner	R12	Sponsoring
R14	Marketing Team	R15	Event Structure
R15	Marketing Team	R14	Reporting
R16	Payment Provider	R4 - R18 - R19 - R20 - R25	Payment
R17	Payment Provider	R8	Payment
R18	Payment Provider	R3 - R4 - R16 - R19 - R20	Payment
R19	Payment Provider	R4 - R16 - R18	Payment
R20	Tax Agencies	R16 - R18	Reporting
R21	Developers	R24 - R25	Database
R22	Developers	Independent	Reporting
R23	Developers	R3 - R4 - R24	Ticketing
R24	Developers	R21 - R23	Database
R25	Developers	R16 - R21	Database
R26	Developers	Independent	General
R27	Developers	Independent	General

TRACIBILITY MATRIX form.

ID	R2	R3	R4	R5	R6	R7	R8	R10	R12	R13	R14	R15	R16	R17	R18	R19	R20	R21	R23	R24
R2					D															
R3			D			D	D	D							D				D	
R4		D											D		D	D			D	
R5					D															
R6	D			D																
R7		D					D													
R8		D				D								D						
R10		D																		
R12										D										
R13										D										
R14												D								
R15											D									
R16			D												D	D	D			
R17							D										D			
R18		D	D									D				D	D			
R19			D									D			D					
R20												D			D					
R21																				D
R23	D		D																	D
R24																		D	D	

In the previous page, the tracibility matrix is more detailed, readable, user friendly, but we also wanted to add dependency information matrix, provided on our course textbook.

MAIN USE CASES

USE CASE 1: Create & Manage Event

Actors

- **Event Manager**

Inputs

- **Event Details:** name, description, date and time of the event
- **Venue:** choosing a venue for the event
- **Ticket:** pricing and types
- **Sale Date:** choosing when to open and when to close ticket sales

Action

1. Organizer inputs event and ticket details, chooses venue and sale dates.
2. System validates input data.
3. System creates and saves the event.
4. Owner of the choosed venue gets a notification from the system.
5. Venue owner accepts or declines the event request.
6. Event manager gets a notification from the system from venue owner.
7. Event manager chooses an auto publish date for the event or publishes immediately.
8. If event manager choosed an auto date, when date satisfied, system publishes the event.

Exceptions

E1 - Step 2 - If input data is incomplete or incorrect, the system displays an error message and prompts the manager to correct the information; goes back to step 1.

E2 - Step 6 - If venue owner declined the event request, the system informs event manager with a refuse notification and lets manager update venue preference; goes back to step 1.

Outputs

- **Event Identifier:** unique code for the event
- **Confirmation Message:** success mesage

MAIN USE CASES cont.

USE CASE 2: Attendee Registration and Ticket Purchase

Actors

- **Event Attendee**

Inputs

- **Attendee Personal Information:** name, contact details
- **Ticket Type:** ticket selection and quantity
- **Payment Info:** choosing payment type and typing payment information

Action

1. Attendee inputs personal information.
2. Attendee selects desired tickets and provides quantity.
3. Attendee provides payment details.
4. System validates input data.
5. System processes payment.
6. If payment succesful, system generates electronic tickets and QR codes for the tickets.
7. Payment system informs attendee with an e-receipt of the purchase.

Exceptions

E1 - Step 4 - If input data is incomplete or incorrect, the system displays an error message and prompts the attendee to correct the information; goes back to step 1.

E2 - Step 6 - If payment fails, the system displays an error message and prompts the attendee to retry or choose an alternative payment method; goes back to step 5.

Outputs

- **Confirmation Message:** succesful registration and ticket purchase
- **Electronic Tickets:** electronic ticket with unique identifiers
- **QR Code:** QR code for the ticket

MAIN USE CASES cont.

USE CASE 3: Onsite Check-In

Actors

- **Venue Staff**

Inputs

- **Attendee QR Code or Ticket Identifier:** QR code or ticket information

Action

1. Event staff scans attendee's QR code or manually enters ticket identification details.
2. System validates the information.
3. System informs event staff about validity of the ticket.
4. System updates attendance status.

Exceptions

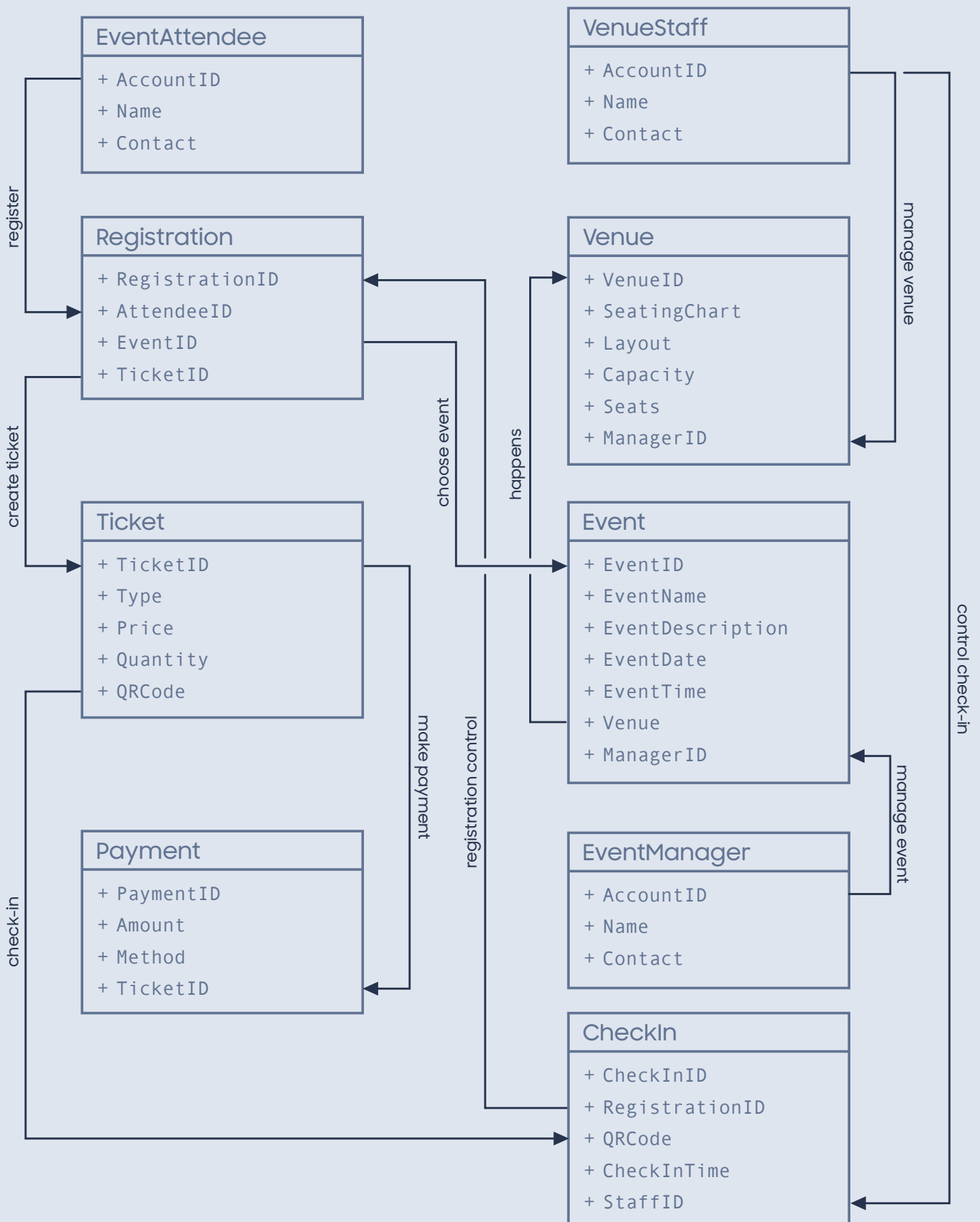
E1 - Step 2 - If input data is incomplete or incorrect, the system displays an error message and prompts the staff to correct the information; goes back to step 1.

E2 - Step 3 - If QR code or ticket identification detail's can't be validate, system informs the staff about incorrect ticket; goes back to step 1.

Outputs

- **Attendance Status:** updating attendance
- **Confirmation Message:** successful check-in

DOMAIN MODEL



CONCLUSION

Gettick, the event management aystem aims to revolutionize event planning and ticketing, providing a comprehensive platform for organizers and attendees.

Through careful planning of requirements, we are confident in delivering a high-quality system that meets the desired expectations.

(!) Some of the requirements doesn't really valid for this particular school project, but we thought like we are doing a real life software project, planning and analysis; that's why we included every aspect.



GETTICK
NEW ERA OF
TICKET & EVENT
MANAGEMENT