Sound Vibes Festival

**Afbeelding met cd

Beschrijving is gegenereerd met hoge betrouwbaarheid**

**Team members:**

Mateo Iliev

Angel Mishkov

Blendi Kovani

Imran Touqi

Table of Contents

[Introduction 4](#_Toc528259608)

[The client 4](#_Toc528259609)

[Processes 5](#_Toc528259610)

[USE CASES 5](#_Toc528259611)

[Scenario 1: PURCHASING TICKETS 5](#_Toc528259612)

[SCENARIO 2: PAYMENT PAGE 6](#_Toc528259613)

[Scenario 3: ADDING FUNDS TO THE ACCOUNT 7](#_Toc528259614)

[Scenario 4: CHECK-IN IN THE EVENT 8](#_Toc528259615)

[. 9](#_Toc528259616)

[Scenario 5: CAMP SITE ENTRANCE 9](#_Toc528259617)

[Scenario 6: PURCHASING FOODS & DRINKS 10](#_Toc528259618)

[Scenario 7: BORROWING EQUIPMENT 11](#_Toc528259619)

[Scenario 8: CHECK OUT FROM THE EVENT 12](#_Toc528259620)

[Functional requirments 12](#_Toc528259621)

[I. APPLICATION MoSCoW 12](#_Toc528259622)

[Must have: 12](#_Toc528259623)

[Should have: 12](#_Toc528259624)

[Could have: 12](#_Toc528259625)

[Wont have: 12](#_Toc528259626)

[II. WEBSITE MoSCoW 13](#_Toc528259627)

[Must have: 13](#_Toc528259628)

[Should have: 13](#_Toc528259629)

[Could have: 13](#_Toc528259630)

[Wont have: 13](#_Toc528259631)

[GUI Applications 14](#_Toc528259632)

[CHECK-IN APPLICATION 14](#_Toc528259633)

[BUY TICKET APPLICATION 15](#_Toc528259634)

[CHECKOUT APPLICATION 16](#_Toc528259635)

[ACCOMODATION APPLICATION 17](#_Toc528259636)

[PRODUCTS FOR SALE APPLICATION 18](#_Toc528259637)

[ITEMS LOAN APPLICATION 19](#_Toc528259638)

[EVENT STATUS APPLICATION 20](#_Toc528259639)

[Event status Guests 20](#_Toc528259640)

[Events Status Shop Statistics 21](#_Toc528259641)

[Events Status Item Statistics 22](#_Toc528259642)

[Website Wireframe 23](#_Toc528259643)

[Database Design 25](#_Toc528259644)

[DESCRIPTION 25](#_Toc528259645)

# Introduction

*This is a set up document for a festival named “Sound vibes”.*

- “The client” section that includes all agreements made with the client.

- “Processes” section that describes all meaningful interactions between the user and the system.

- “Functional requirements” section, including all the requirements for the system, grouped by application/website and prioritized by MoSCoW.

- “GUI” section giving a brief description of the design of the application and how it works.

- “Website Wireframe” section, showing the design of the website and brief description of it.

- “Database Design” section, showing the design of the database and a brief description of it.

# The client

The client’s name is Matthijs Kuiper he is a representative from a company which specializes in organizing festivals.

We made a verbal agreement with the client to provide functional website from which you could buy tickets, applications regarding exchange of money for goods, application showing the status of the event and an application for checking in and out.

# Processes

## USE CASES

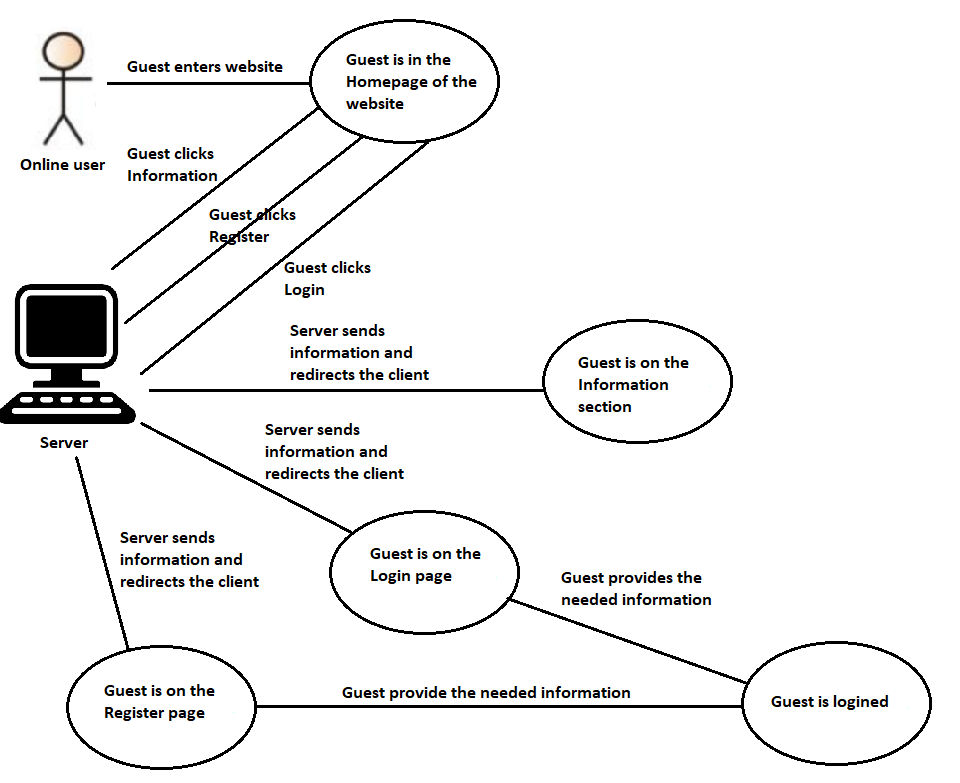
### Scenario 1: PURCHASING TICKETS

Role: Online user

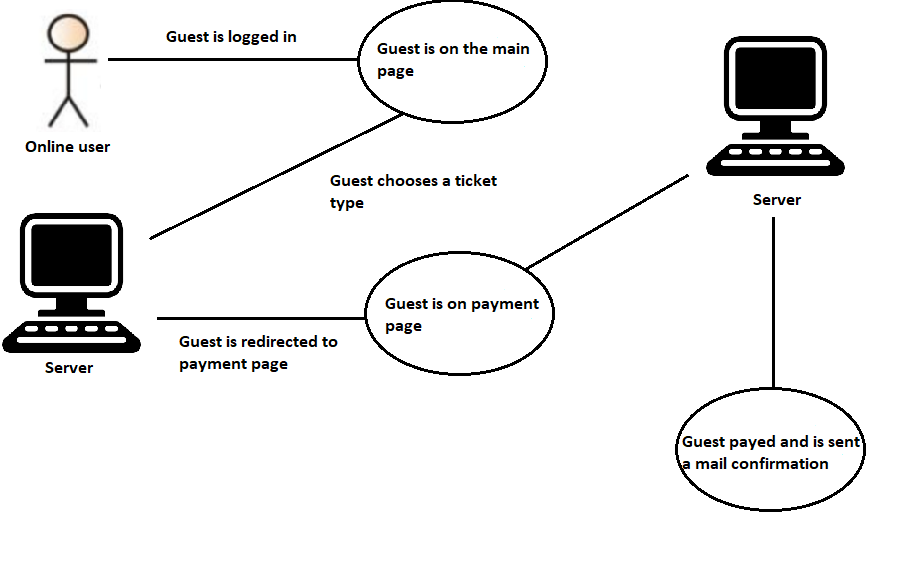
**Prerequisite:** The user was searching his social media and saw an ad about the festival

1. The guest enters the website
2. The guest is in the Homepage
3. The guest clicks on “Information” to find out more about the festival
4. The guest clicks “Register” and register page is opened
5. The guest provides needed information such as- email, username and password
6. The guest submits the information
7. The guest clicks “Login” and login page is opened
8. The guest chooses a ticket type
9. The guest confirms the ticket type and is directed to payment page
10. The guest enters his payment details
11. Payment is completed
12. The guest is redirected to the main page
13. Send mail with confirmation

Logged in:



Buy tickets:



**Possible exceptions:**

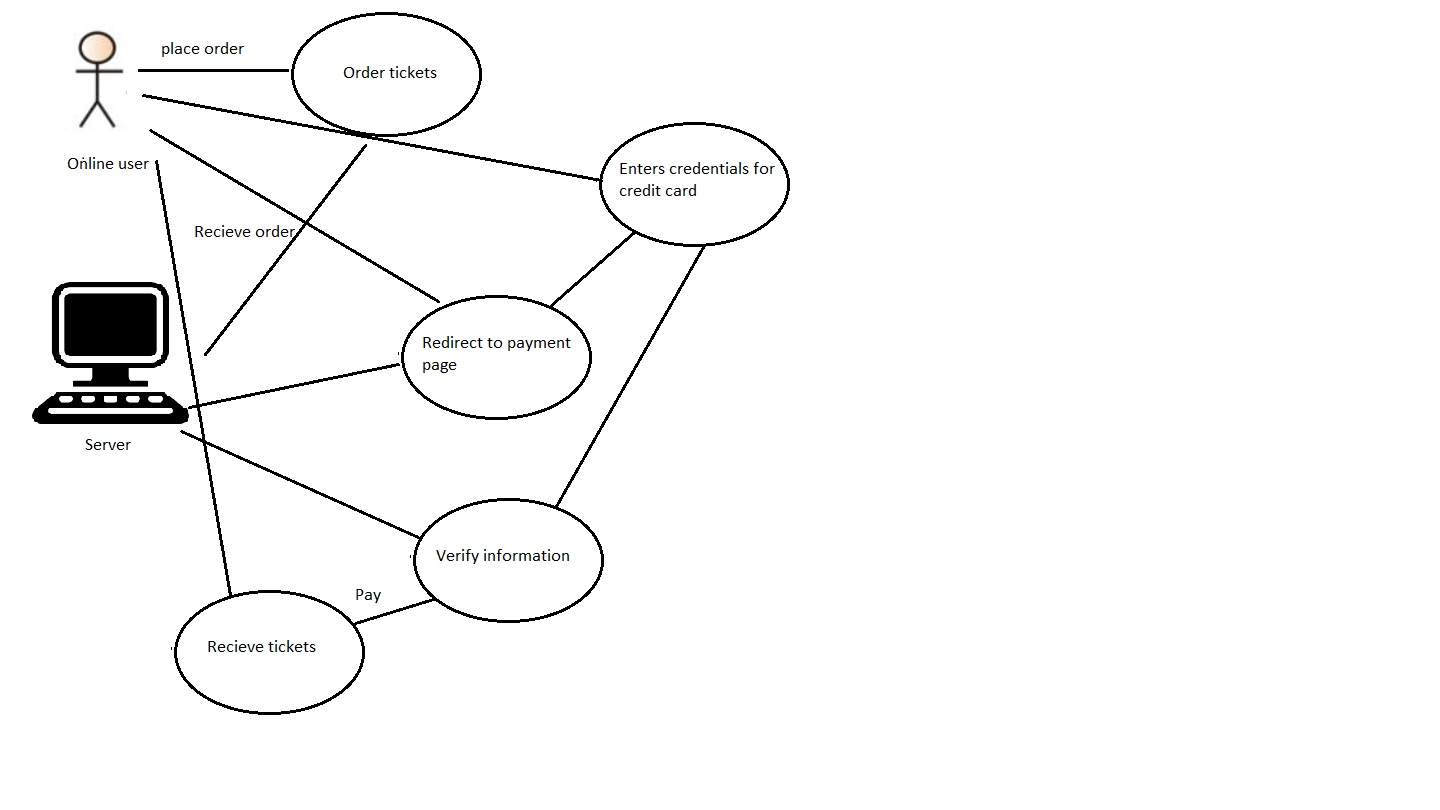
1. The user name is already taken, the system displays” Username is taken!”, the guest choses new username and proceed to submit information.
2. The user inputs the wrong type of data the system displays” Please input correctly all of the fields!”
3. The email is already used the system displays” The email is taken!” the user is required to enter a new email.”

### SCENARIO 2: PAYMENT PAGE

Role: Online user

**Prerequisite:** The user is at the payment page after he has chosen the type of ticket he wants

1. The guest is at the payment page
2. The guest enters the credentials asked from his such as – name, CCV, 16-digit code from the credit card and expiration date
3. The guest then presses “Process”
4. The payment is done
5. The guest is returned to the main page



**Possible exceptions:**

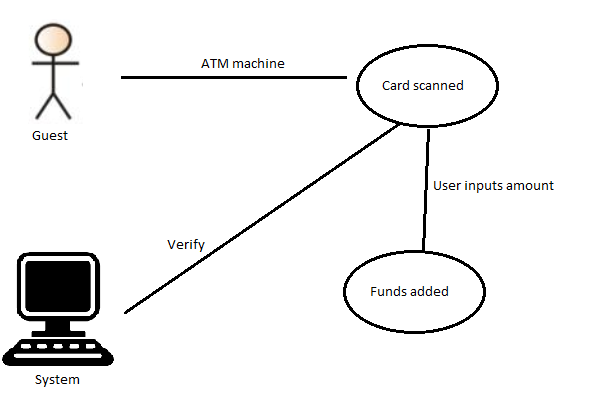
1. The user enters in the 16-digit code some letters
2. The user’s credit card has expired

### Scenario 3: ADDING FUNDS TO THE ACCOUNT

Role: Guest

**Prerequisite:** The guest is at the event, on the machine for adding funds

1. The guest scans his RFID card
2. The guest enters the amount of money he wants to put in his account
3. The guest press “Complete”
4. The inputted amount is added to his account



**Possible exceptions:**

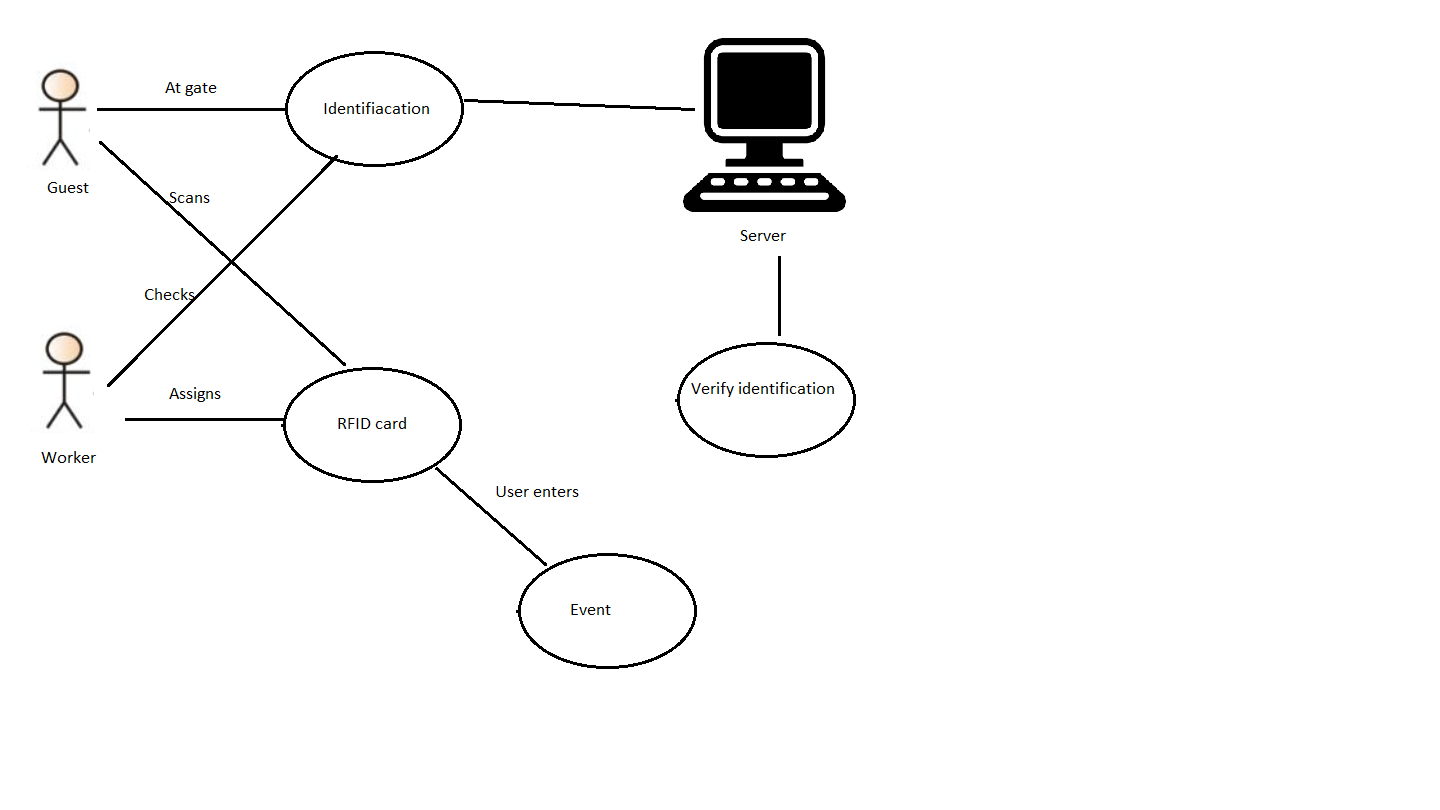
1. The user enters invalid amount, the system displays “Invalid amount, please try again!” then the user enters a valid amount.

### Scenario 4: CHECK-IN IN THE EVENT

Role: Employee, User

**Prerequisite:** The user is at the event and he is at the gate waiting to be checked in by employee.

1. The user is at the gate
2. The user is asked for identification.
3. The worker types in his identification document in the system.
4. The worker takes a card and assigns the ID number of the card to the user
5. The guest then scans the card and enters the event



**Possible exceptions:**

1. The user doesn’t have a ticket, he is redirected to the machine to buy a ticket

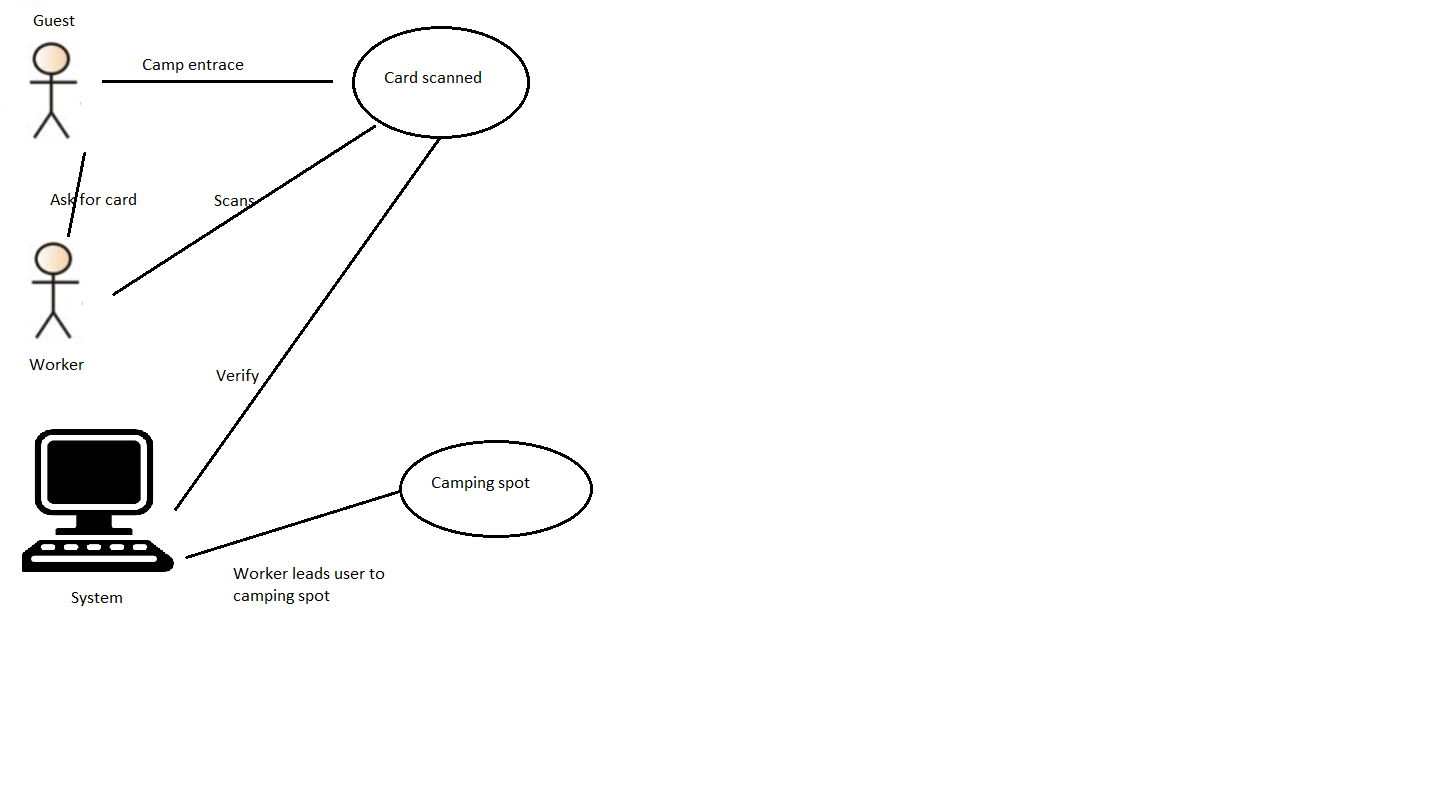
### .

### Scenario 5: CAMP SITE ENTRANCE

Role: User

**Prerequisite:** The user is at the entrance of the camp site and wants to go to his camp spot. He needs to be checked in.

1. The user is at the camp site entrance
2. The user gets his card scanned
3. Then the system goes through the database to check whether this person has an accommodation
4. The user is guided to his camp spot



**Possible exceptions:**

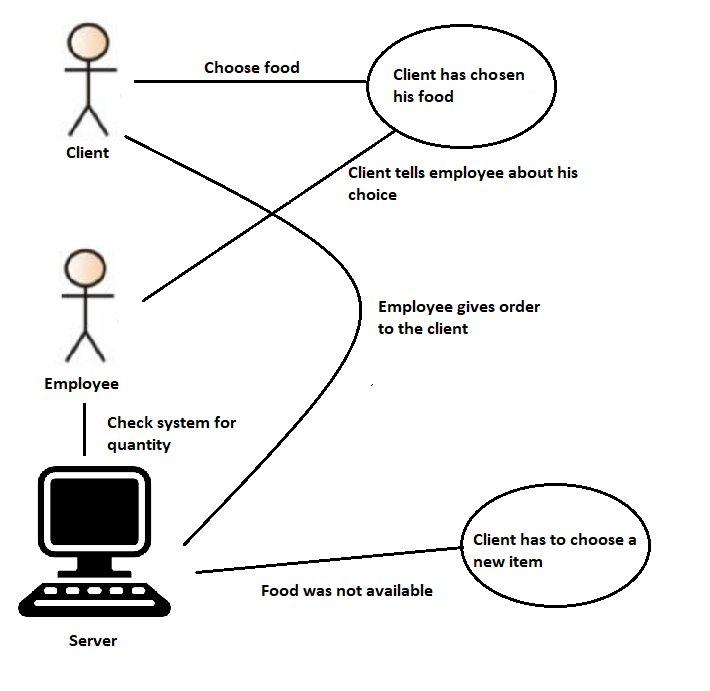
1. The user does not have an accommodation and has not reserved one. He will be asked for the information asked in the website, about reserving a camp spot and then he can pay on spot and be checked in.

### Scenario 6: PURCHASING FOODS & DRINKS

Role: User, Employee

**Prerequisite:** The user is already checked-in in the event and now he is looking to buy food. He is standing in front of the food stand.

1. The user is looking at his options at the food stand and chooses a product.
2. The employee checks the system if that product is in it.
3. The employee scans the card of the user
4. The user pays with his account
5. The goods are exchanged



**Possible exceptions:**

1. The user does not have enough balance in his account, and he is sent to deposit money in his account.

### Scenario 7: BORROWING EQUIPMENT

Role: User, Employee

**Prerequisite:** The user wants to borrow some equipment, he is looking for the equipment stand, so that he can show some video footage to his friends.

1. The user is at the stand
2. The employee checks the system for the product.
3. The employee then scans the card of the user
4. The user says for how long he wants to borrow the item for
5. Then the user pays with his card
6. The goods are exchanged

**Possible exceptions:**

1. The user does not have enough balance in his account, and he is sent to deposit money in his account.

### Scenario 8: CHECK OUT FROM THE EVENT

Role: User, Employee

**Prerequisite:** The user is at the exit gate and wants to leave the festival.

1. The user scans his card
2. The employee checks system if they have returned all items borrowed
3. The user is checked out and leaves the festival

**Possible exceptions:**

1. The user has not returned all of the borrowed items and he is asked to return before leaving.

# Functional requirments

## APPLICATION MoSCoW

### Must have:

* Check-in and out function
* Borrowing items from shops
* Deposit money at the ATM
* Show info for all the money deposited
* Show free camping spots
* Show person details
* Rent a camping spot, if there is one available
* Buy ticket at the entrance

### Should have:

* Show transaction history of purchases

### Could have:

* Total amount of food and drinks sold

### Wont have:

* The statistic application will not have mobile version

## WEBSITE MoSCoW

### Must have:

* Option to buy tickets
* Show information about performers
* Reserve a camp spot

### Should have:

* A very detailed information about the event
* Ticket information

### Could have:

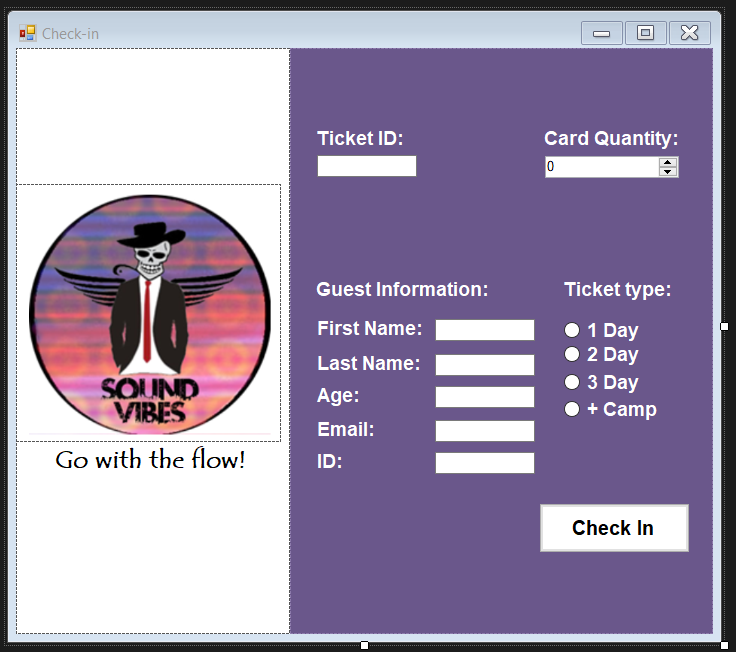
* Ticket history

### Wont have:

* Pictures from previous events

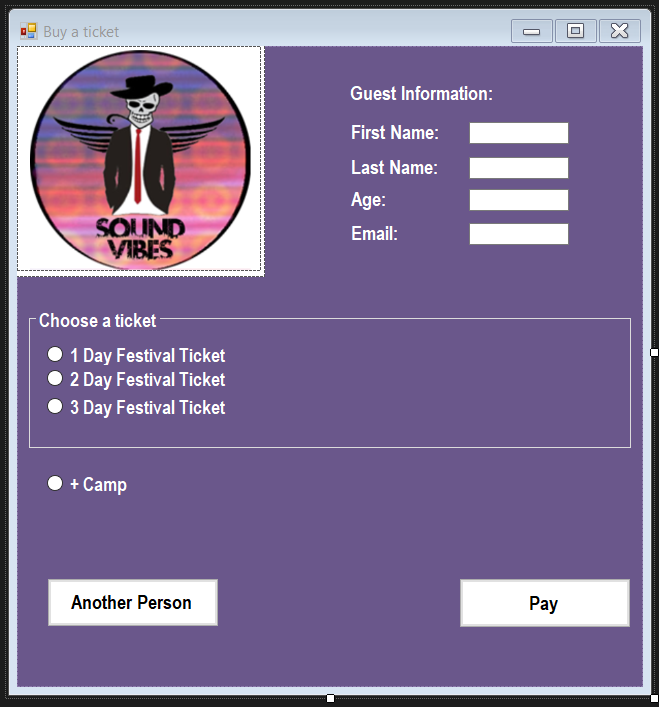
# GUI Applications

## CHECK-IN APPLICATION



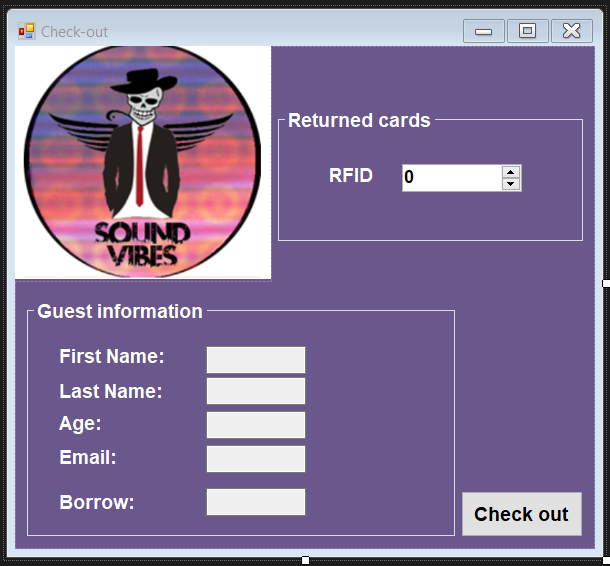
1. Employee opens the “Check-in application”
2. The guest scan his bracelet
3. Data appears in the text boxes and the ticket type is selected as well
4. Employee asks for verification from the guest
5. Employee gives the amount of bracelets required from the guest
6. Employee clicks “Check-in”
7. User is registered in the database

## BUY TICKET APPLICATION



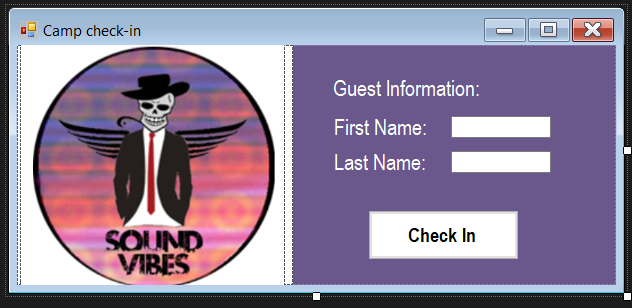
1. Guest on machine for ticket purchase
2. System ask for details
3. The guest fills in the text box fields
4. The guest chooses ticket type and decides if he wants a tent or no
5. The guest has the option to add another ticket if he wants to buy more than one
6. The guest clicks on “Pay”
7. New window opens “Payment”
8. The user pays and a new event account is created

## CHECKOUT APPLICATION



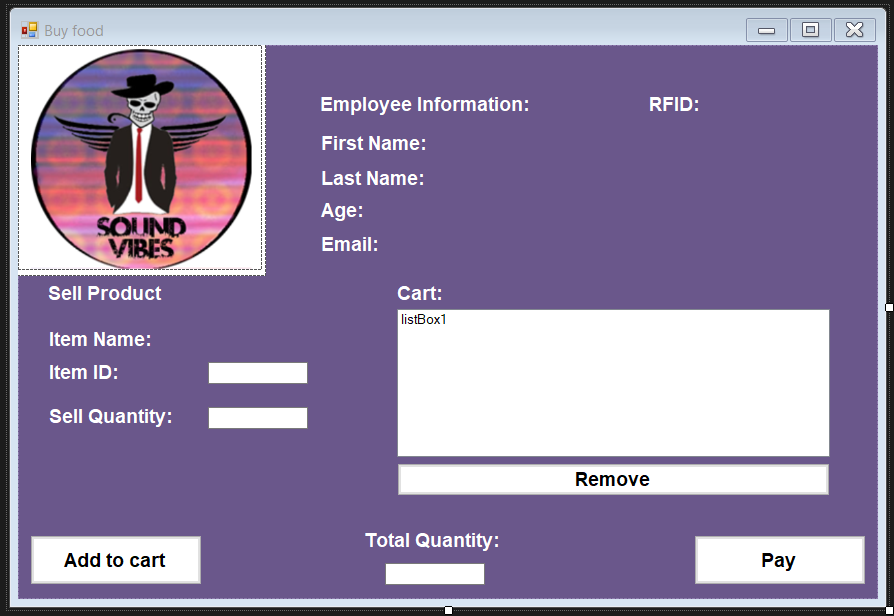
1. Guest is at the checkout gate
2. Guest scans his bracelet
3. Application shows all of the information about the guest
4. The system displays if he has to return an item or no
5. Employee fills in how many bracelets the guest is returning
6. Employee scans the bracelet again and then “check out”
7. The guest’s personal information is removed from the database

## ACCOMODATION APPLICATION



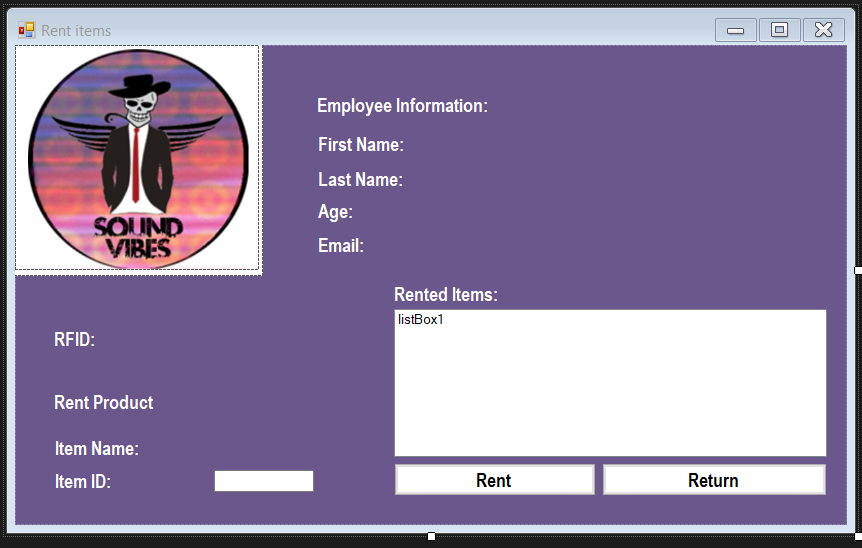
1. Guest scans his bracelet
2. Information shows in the system
3. Employee checks identification
4. Employee clicks on “Check-in” and guest is checked in

## PRODUCTS FOR SALE APPLICATION



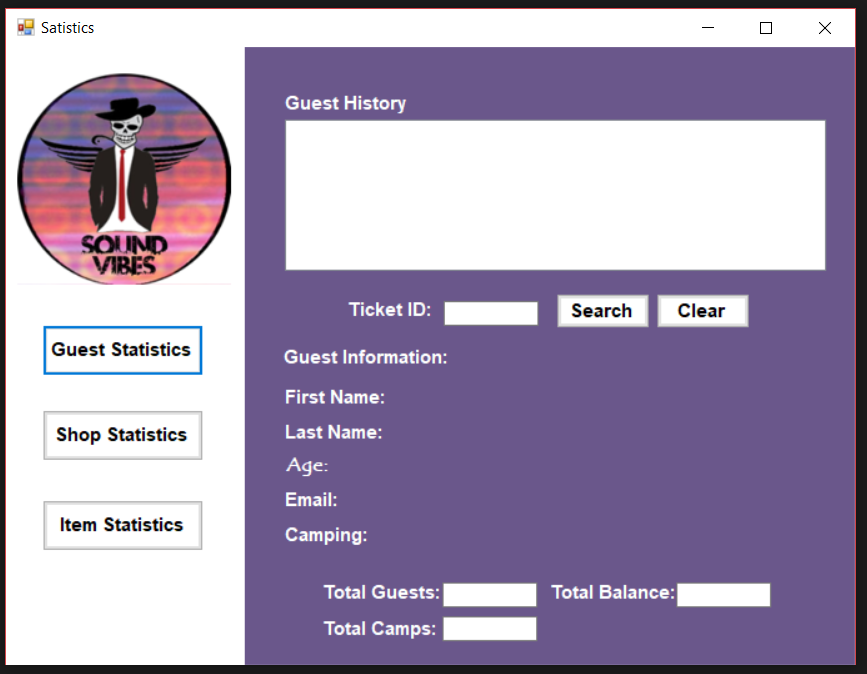
1. The guest chooses what he wants
2. Employee presses the icon with the selected choice
3. Then the stock is shown on the right side “Total Quantity”
4. The employee enters the desired quantity from the guest in the “Sell Quantity” text box
5. The employee add the items to the card
6. If the guest changes his mind he can remove the item from the cart
7. The employee presses the “Pay” button
8. The guest scans his bracelet

## ITEMS LOAN APPLICATION



1. The guest chooses what he wants
2. Employee presses the icon with the selected choice
3. Then the stock is shown on the right side “Total Quantity”
4. The employee enters the desired quantity from the guest in the “Sell Quantity” text box
5. The employee add the items to the card
6. If the guest changes his mind he can remove the item from the cart
7. The employee presses the “Pay” button
8. The guest scans his bracelet

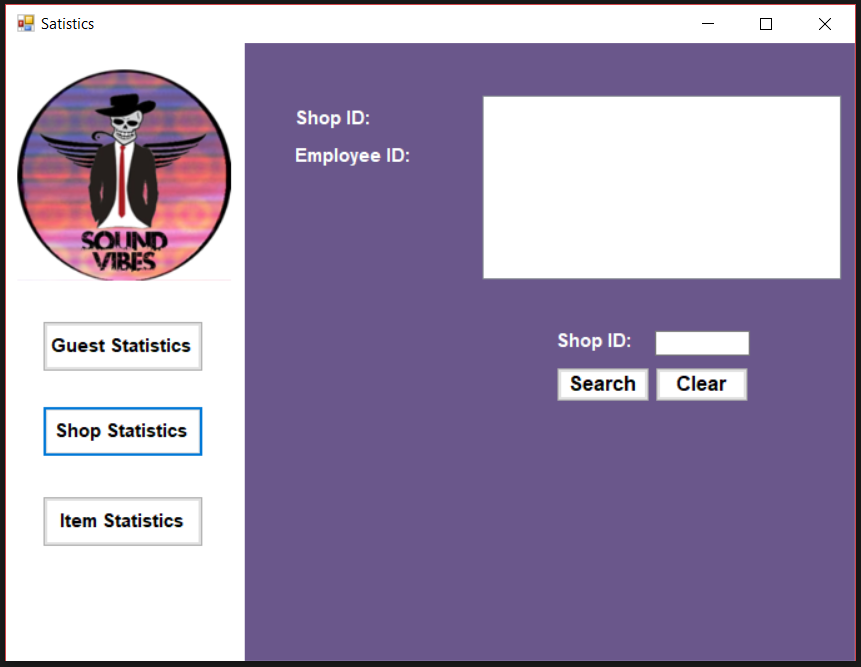
## EVENT STATUS APPLICATION



### Event status Guests

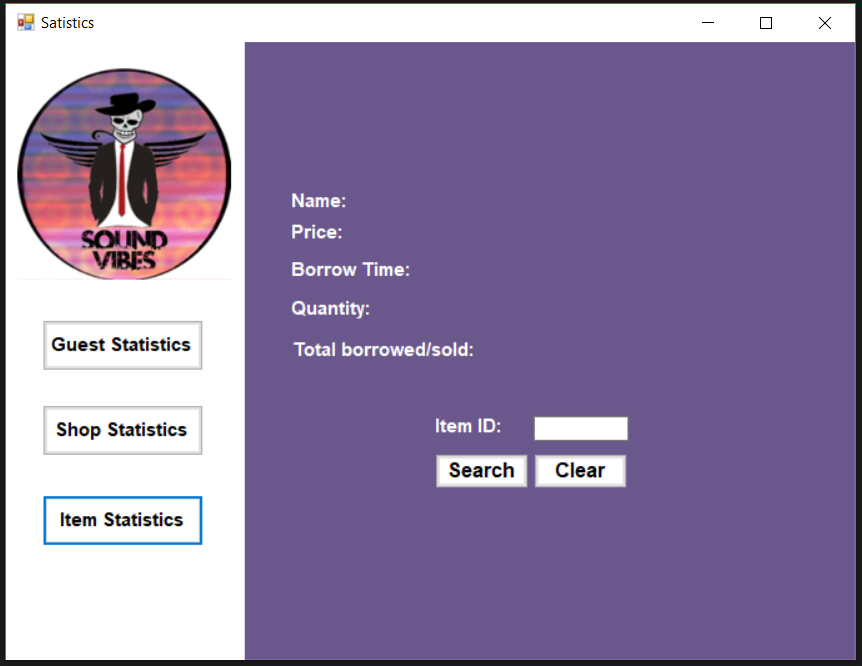
The manager opens the “Event status application”

1. The manger selects “Guests”
2. The manger fills the Ticket ID of the guest
3. Then clicks on “Search”
4. All the data is shown on the window
5. All the purchase history of the guest appears in the “Guest History”
6. In the “Total guests”, “Total camps”, “Total balance” appears the data for all of the guests that have ticket and that are checked in the event, all camping spots left and total balance for all of the events accounts



### Events Status Shop Statistics

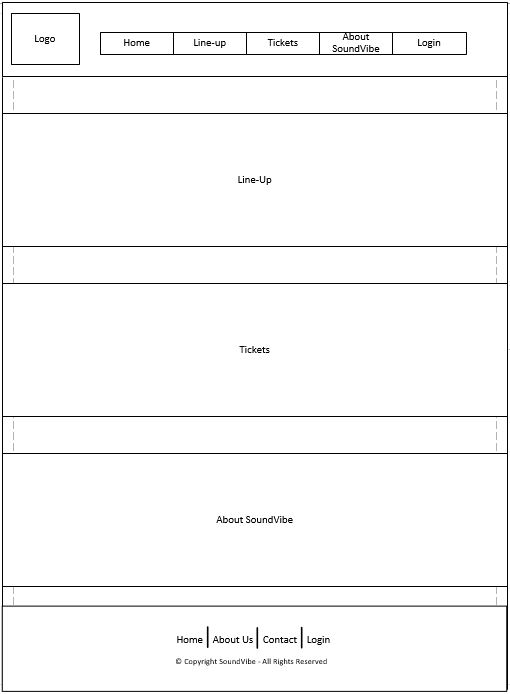
1. The manager presses on “Shop statistics”
2. The manager fills in the “Shop ID” he is interested in
3. All the data about the shop appears in the white box
4. All the employee ID’s that are working in that shop will be present
5. The manager has the option to “Clear” the data as well

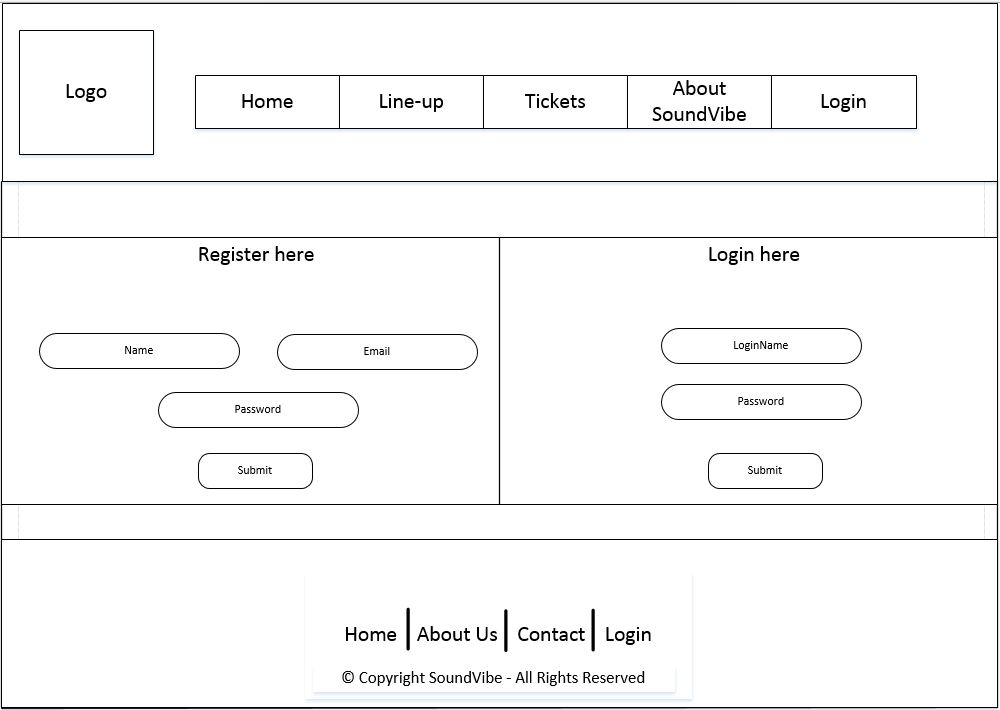
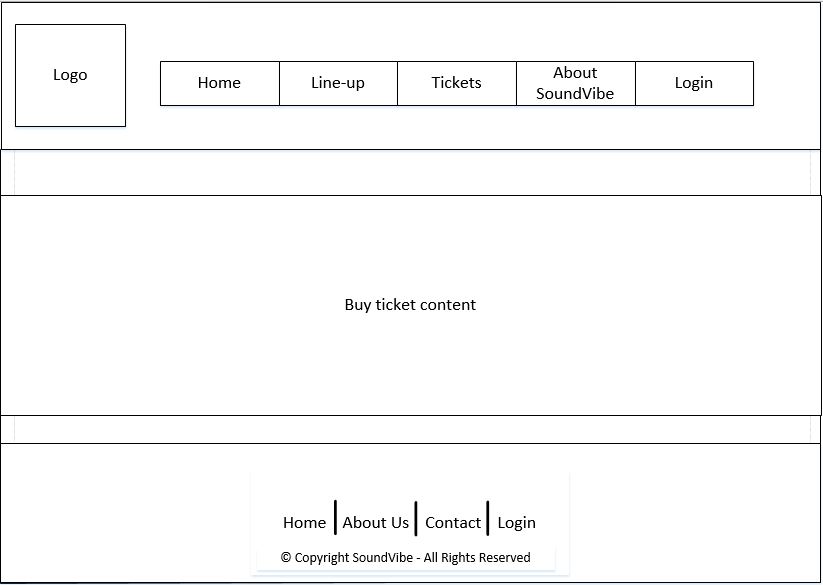


### Events Status Item Statistics

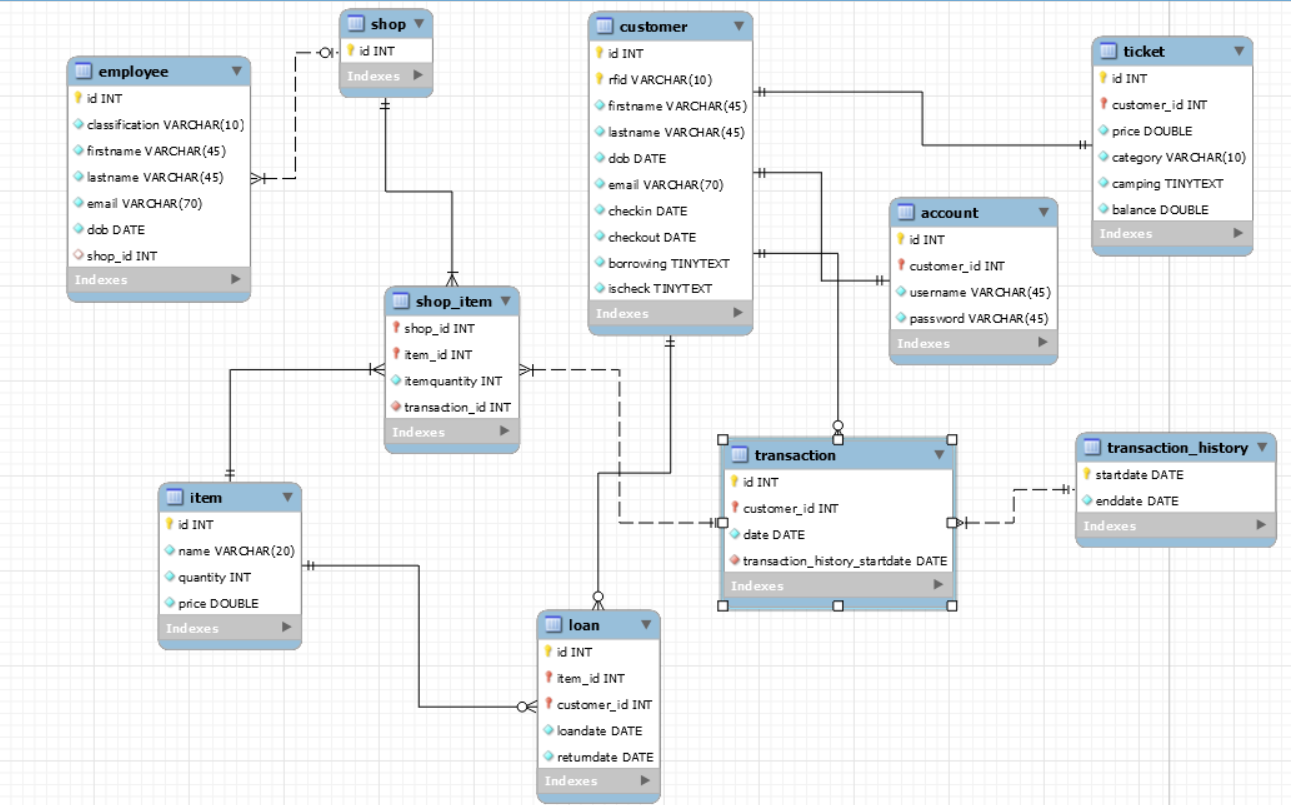
1. The manager presses on “Item Statistics”
2. The manager fills in the “Product ID” he is interested in
3. The manager then clicks on “Search”
4. All of the information about the item appears
5. The borrow date shown is the last borrow date of this product.
6. The manager has the option to “Clear” the data as well

# Website Wireframe





# Database Design



## DESCRIPTION

Every shopwould have a specific id number, employees can work at the shops but only assigned to one shop at a time and many employees can work at one shop.

Customers would have a ticket, also showing its category and if camping is included or not. And with every ticket it comes with an account where the customer can store money in it during the event.

Customers can also borrow some items during the event, each loan with its own id loan date and return date.

Items could be lent or can be bought with a price, the shop item table stores information about every product included in the shop and when the customer wants to buy the item a unique transaction is then created. The transaction tables store information about when the product was bought. This would even be passed down to the transaction history where multiple transactions would be stored of the same customer.