

**Setup document**

**Organizing the “Kuierpad WinterFest” event**

Course: ProP

Group: 6 “TechQuest”

**Team**:

Kien Duc Do - 3623941

Andreea Dumitru - 3479943

Atanas Draganov - 3508277

Abdullah Alkhalaf- 3579506

**Table of content**

|  |
| --- |
| 1. Preface 2. Agreements – made with the client 3. Processes   Use-cases:  3.1 Buying tickets  3.2 Sign in  3.3 Write a review  3.4 Deposit money  3.5 Purchasing items using the QR code  3.6 Renting items  3.7 Camping spot reservation   1. Functional requirements   4.1 Website  4.2 Application   1. GUI – to do 2. Website & wireframe 3. ERD |

**1)Preface**

The document contains details of the agreements between the client Mr Frank de Lepper and the team of TechQuest for the creation of the software for the event WinterFest.

In this setup document we will talk through the deliverables as described in the project plan. We will describe the processes, functionalities and user interfaces for every application TechQuest will build.

**2)Agreements – made with the client**

We agreed to deliver the following software for the event organised by our client Mr Frank de Lepper.

**Applications:**

- An application to be used at the entrance/exit of the event;

- An application to be used at the entrance/exit of the campsite;

- An application to be used at the shops;

- An application to be used at the loaning stands;

- An application for the organization to inspect the status of the event (dashboard);

- An application to be used for the ATM.

**Others:**

- A website;

- A database;

- A presentation about the project;

**Extra features that can be added:**

- Newsletter;

- A refund option at the shops/loaning stands;

- Keep track how much of each item has been sold;

- Show the most sold items in the overview app;

- Employees login with bracelets and automatically opens the GUI of their workplace;

- The option for choosing to sleep in the same camping spot with others.

**3)Processes**

**Preconditions:**

The following preconditions are valid for every use-case mentioned under:

1. The application is running.
2. The user is logged in with their unique username and password.

If there is a precondition unique for an application, it will be mentioned in the specific use-case.

**Exceptions:**

The following exceptions are valid every use-case mentioned under:

1. Connection lost.

**3.1 Buying tickets**

1. The website is accessed;
2. Customer presses the “Tickets” button;
3. The number of tickets is being chosen & number of persons for the camping spot as well;
4. Press “Next” button and go to the step 2 of purchasing the ticket;
5. Fill in with the personal information of the person/s and press “Next”;
6. Now the customer is redirected to the payment page ;
7. [OPTIONAL] The ticket is being downloaded.
8. [OPTIONAL] The customer redirected to the login page.
9. ///!!!!==IMPROVE THE TEST CASES==!!!/////--add info related to the system

**3.2 Sign In**

* + 1. Press the “Sign In” button;
    2. Fill in the username;
    3. Fill in the password;
    4. The user is welcomed and details about his/her account are being shown.

**3.3 Write a review**

* + 1. Access the website;
    2. Click on the “Reviews” button;
    3. If having an account, he/she can review the event. If the person doesn’t have an account, giving a review is possible only after purchasing a ticket.

**3.4 Deposit money**

* + 1. Purchase ticket
    2. Sign in to the personal account
    3. Press the “Deposit” button after filling with the desired amount.

**3.5 Purchasing items using the QR code**

* + 1. Go to the shop;
    2. Choose an item/items;
    3. Open the app [or hand in the QR code];
    4. The employee scans the QR code.
    5. Employee: confirm the purchase.

**3.6 Renting items**

Same procedure as in the case of purchasing items 3.5.

**3.7 Camping spot reservation**

The camping spot reservation is being processed when purchasing the ticket.

**4. Functional requirements**

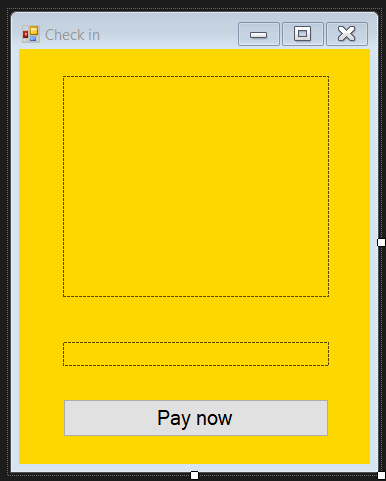
**MoSCoW table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirements** | **Must** | **Should** | **Could** | **Won’t** |
| **4.1 Website** | | | | |
| Check the information about the Event | **☺** |  |  |  |
| Contact with the Organizer. |  | **☺** |  |  |
| Buy tickets and reserve camping spots. | **☺** |  |  |  |
| Refund a ticket and camping spots. |  |  |  | **☺** |
| Register an account for the website. | **☺** |  |  |  |
| Re-book the tickets and camping spots |  |  |  | **☺** |
| Log in with Facebook or Twitter |  |  |  | **☺** |
| Newsletter |  |  | **☺** |  |
| Admin section |  | **☺** |  |  |
| Add balance to the employees |  |  | **☺** |  |
| Log in with the QR code | **☺** |  |  |  |
| **Requirements** | **Must** | **Should** | **Could** | **Won’t** |
| **4.2 Applications** | | | | |
| Entrance/Exit festival application | **☺** |  |  |  |
| Is entry successful? |  | **☺** |  |  |
| Employee: Campsite access? | **☺** |  |  |  |
| User info transferred to QR code | **☺** |  |  |  |
| Employee: Exit festival application |  | **☺** |  |  |
| Employee: Loaned items returned? | **☺** |  |  |  |
| Employee: Food and Drinks shops application | **☺** |  |  |  |
| Employee check in |  |  | **☺** |  |
| Remove an item from the order |  | **☺** |  |  |
| Order preview |  | **☺** |  |  |
| Total cost |  | **☺** |  |  |
| Buying of items is successful? | **☺** |  |  |  |
| Decrease the balance of the user |  | **☺** |  |  |
| Loaning stands application | **☺** |  |  |  |
| Select multiple items |  |  | **☺** |  |
| Order preview |  | **☺** |  |  |
| Loaning successful? |  | **☺** |  |  |
| Employee: Return item | **☺** |  |  |  |
| Update database (delivery) | **☺** |  |  |  |

**5. GUI**

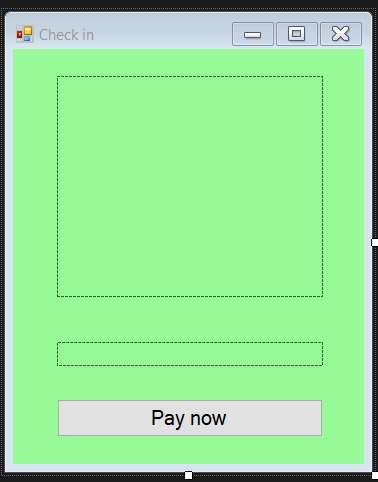
The applications will be used by employees during the event. There are several GUI’s for the different applications.

**For entrance check-in:**



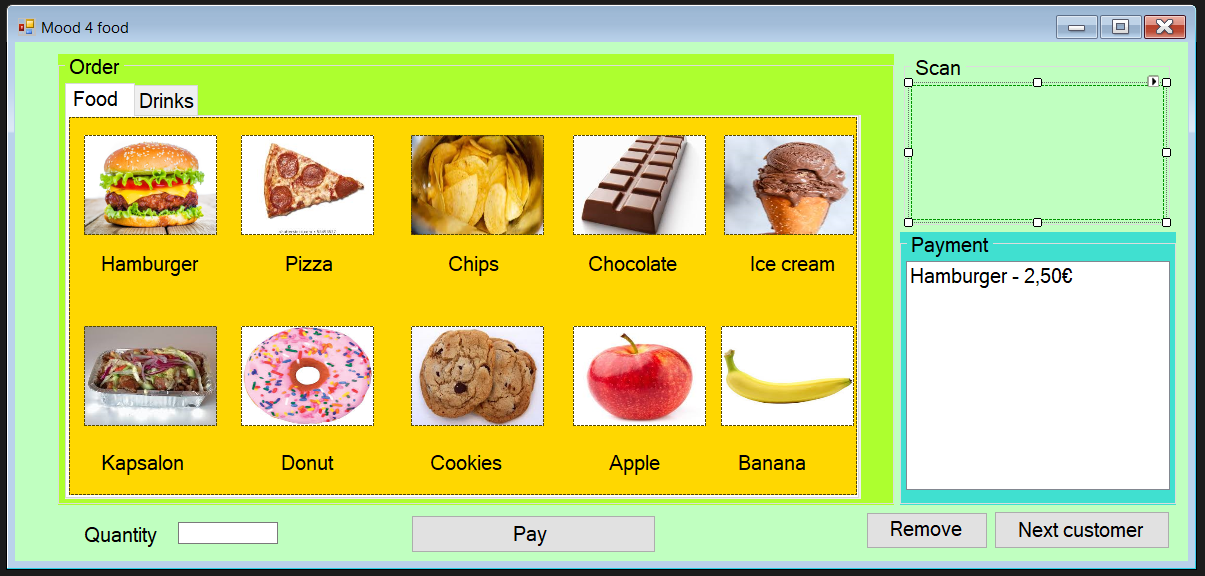
The big picturebox will be used to scan a qr-code while the small one will be used as an indicator(with colors red and green). If the color turns red it means the visitor is not allowed to enter. The button “Pay now” will be visible only if the indicator turns red.

**For camping spots check-in:**



As long as the procedure with checking camping spots is the same, employees will use a similar application for that purpose.

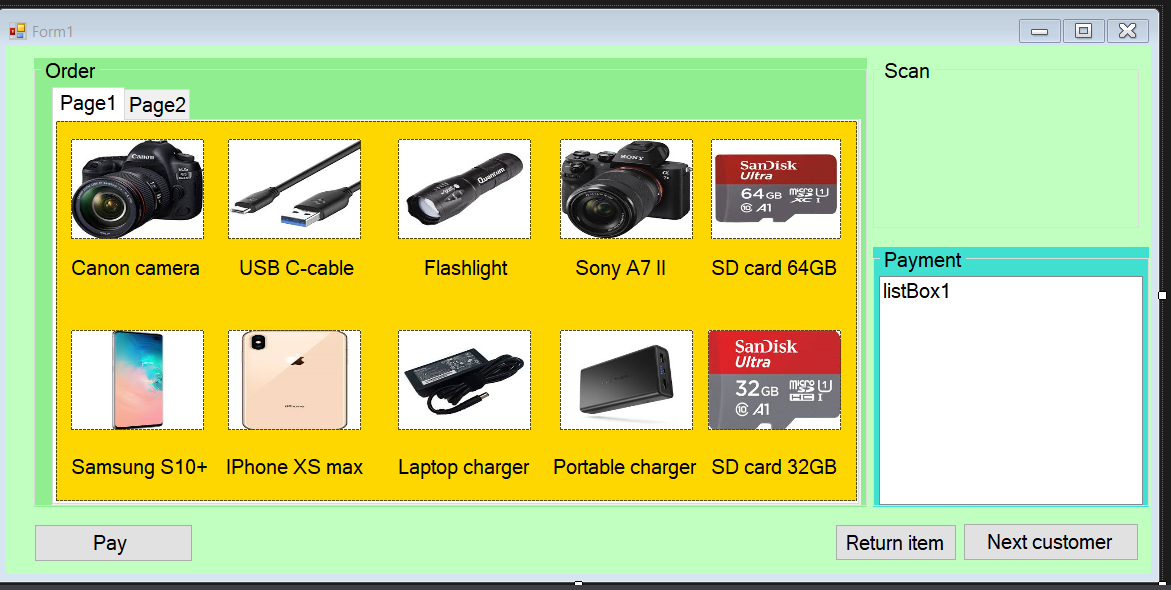
**For the shops:**



The goods are divided in categories “Food” and “Drinks” with 10 items in each category.

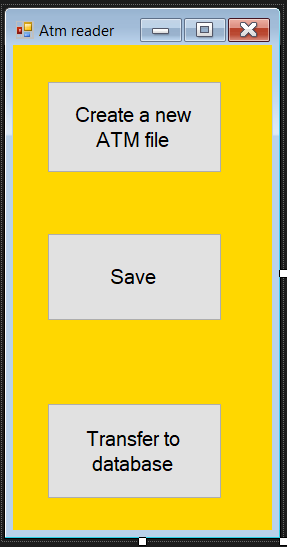
For the ease of the employees if a customer wants only 1 item, a click on the correct picturebox will add the item to the listbox in Payment groupbox. The Quantity textbox can be used in case there is a group of customers, for example. There is a Remove button which is used to remove items which are already in the listbox. The payment is in the end of the procedure with the QR-scanner on the top-right. The button Next customer clears the listbox.

**For loan materials:**



The GUI is almost the same as the one for the shops but there are a few differences. There are not two categories for the items, yet again we have two tabs containing ten items each. Another difference is the Return item which will be used to clear the information for the user of the item from the database.

**For ATM files:**

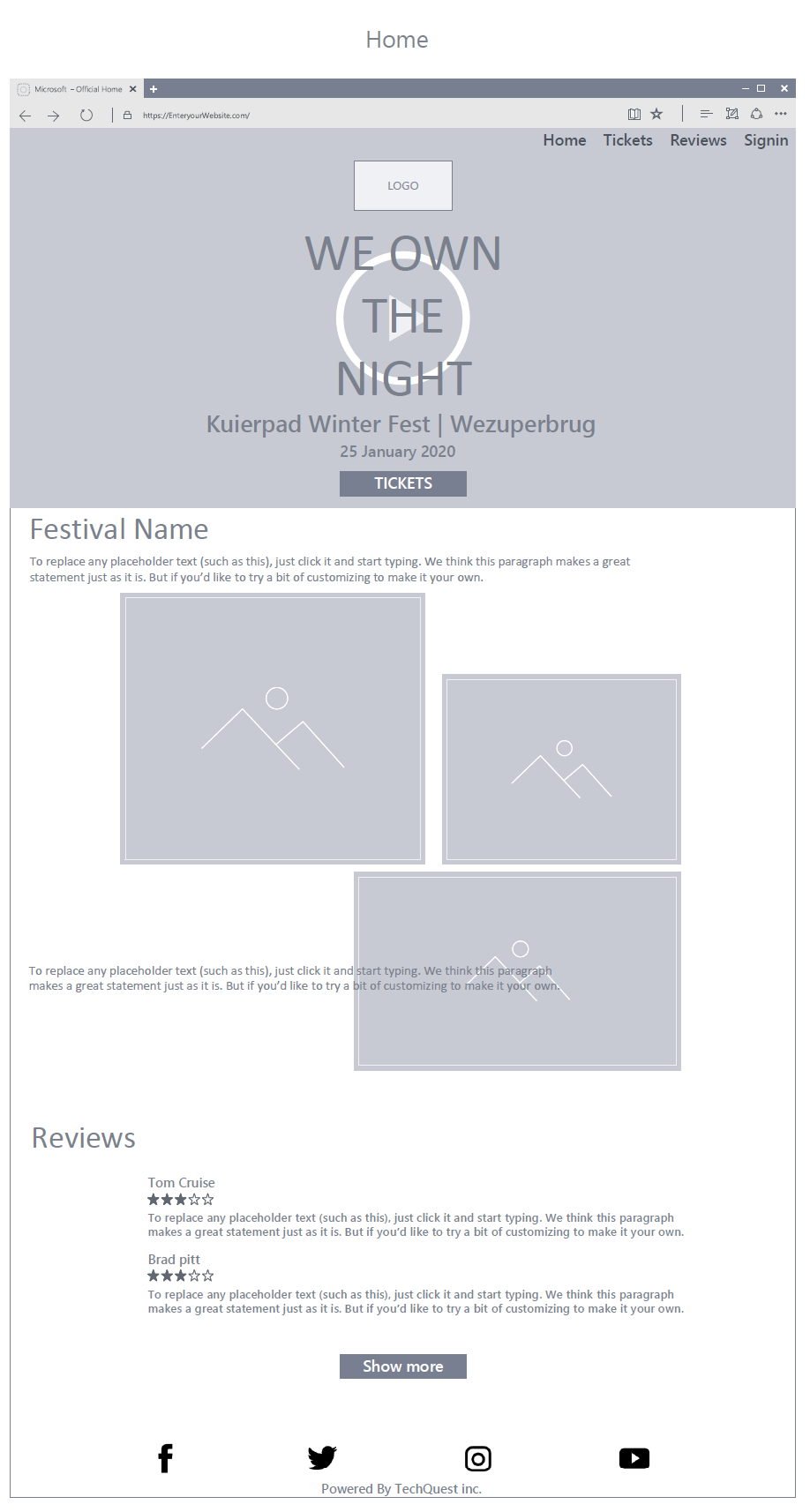


The application called “Atm reader” has a simple but effective GUI with only three buttons. “Create a new ATM file” button will be used to open a directory and to create a text file where the ATM log can be pasted. The “Save button” saves the information and then the “Transfer to database” button is used to convert every line of the ATM file to valuable information and to add it in the database.

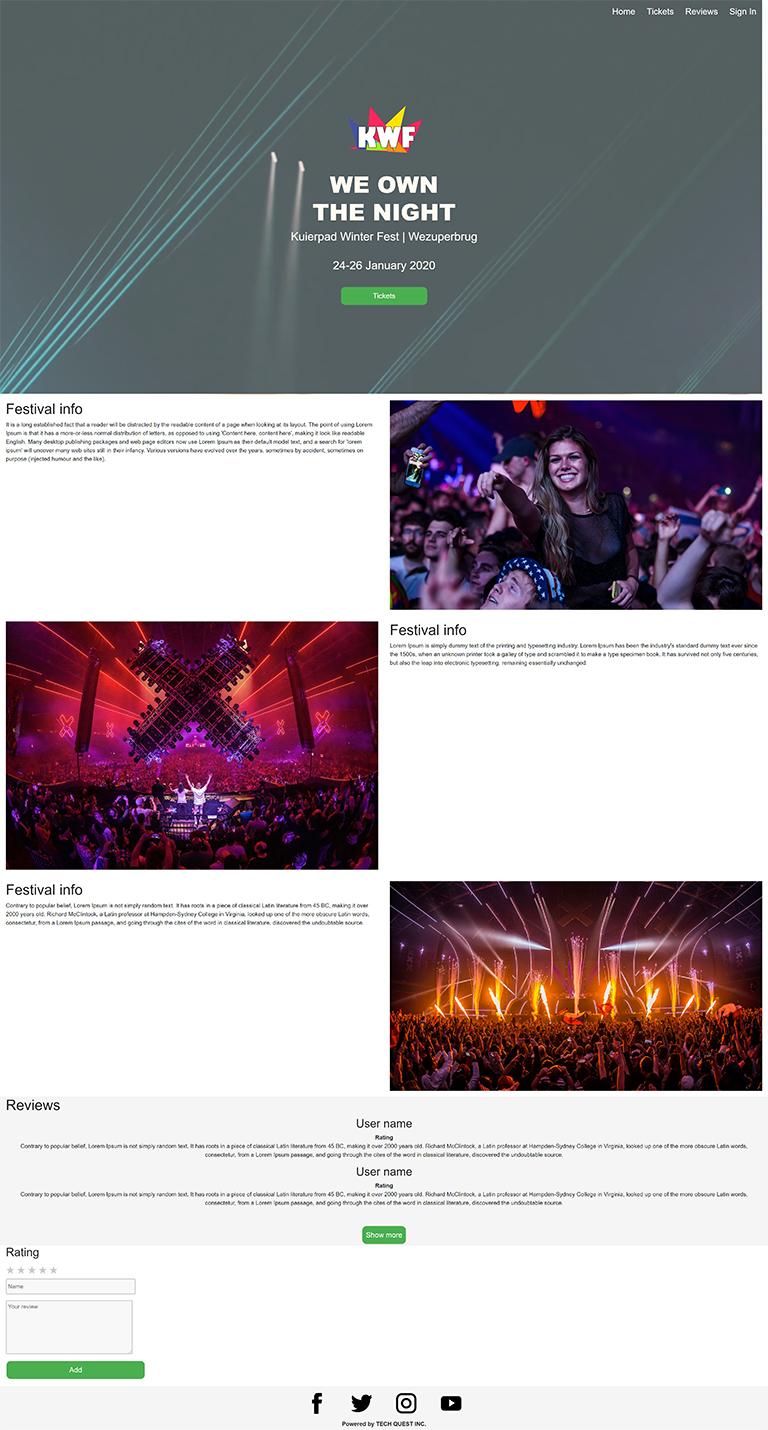
**6. Website & wireframe**

The website will be user-friendly. All the information about the event, login and buying tickets will be easy to find. People will be able to see the line-up and schedule of the event. Information such as tickets bought, access to the event and the camping area, money spent will be available in the account information of each user.

**Home wireframe:**

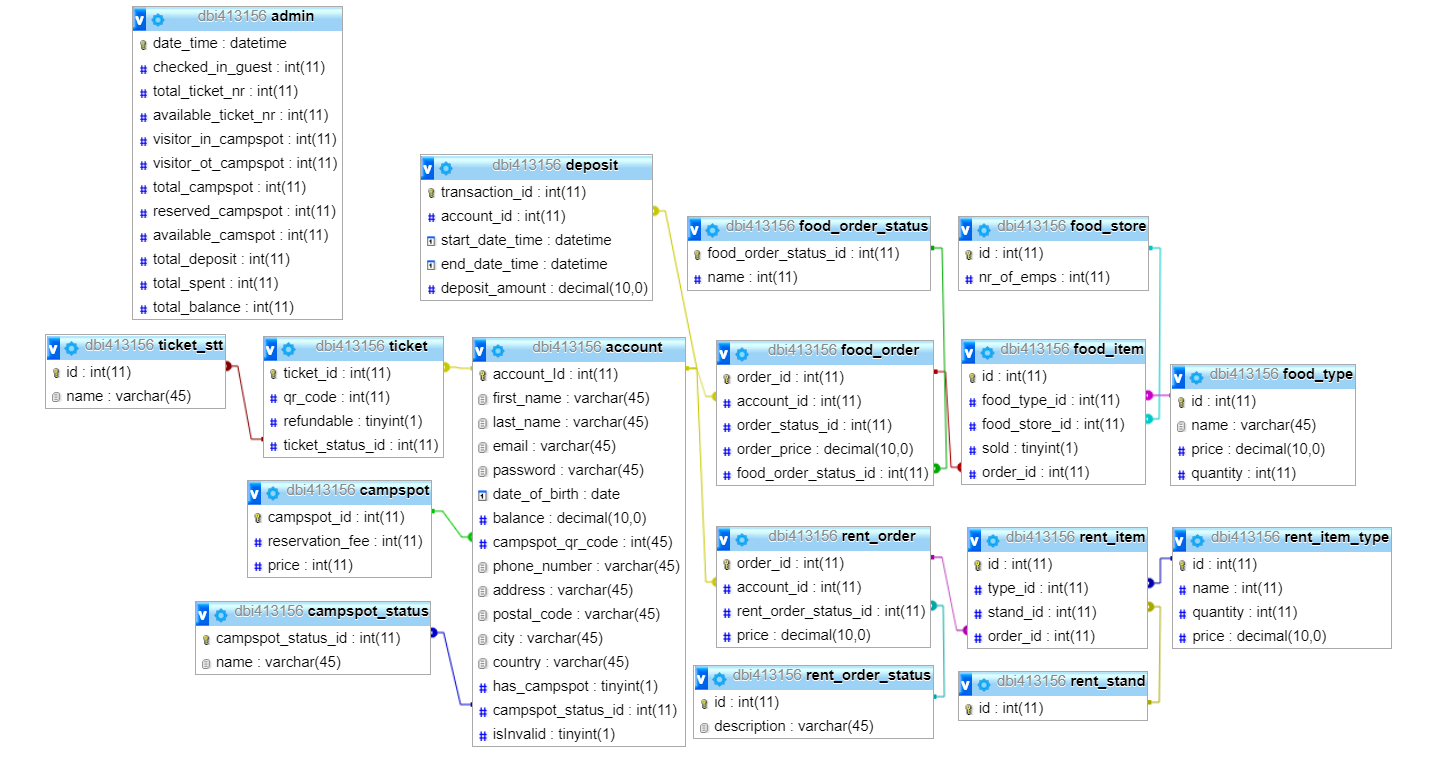


**Home visual design:**

****

**7. ERD**

There are several tables in the database design below. We want each account has a unique ticket. An account can belong to one campspot (possibly none). Ticket and account table has its status table. An account can have many food orders or rent orders. And each of these orders has its status table. For food and drink, there are separate types for each item (similar for renting item). Each item can be found in which store it belongs to, hence food store table and stand table exist. Also each item can be a part of an order from a specific account. And you can also track order status. An account can deposit an amount of money. There is also a table for admin, which stores necessary statistics.

****