Restaurant App Design document

## Purpose of the application

The purpose of this application is giving restaurant owners an easy way to set-up a layout for their restaurant, with this app it will be very easy to create the layout that a specific restaurant owner wants.

## High level entities

### These are the important objects that our application will hold.

Customers

Table

Merged-table

Bar

Waiting area

Group area

Smoking area

## Low level design

### Customers

This is an object that defines the random aspect of the application, a customer is responsible for using the table and merged-table objects. There will be a number on the table which will show the amount of customers that are sitting at that table.

### Component

This is an object used in the creation of tables, merged-tables and the bar objects.

### Table

This is an object groups multiple customer objects together. The table is colored green when it is available, red when occupied. The size can determine the look of the table on the floor plan. A number on the table will show the size of the table.

Table contains a variable for the capacity of the table, in other words, the amount of people that can sit at a table.

### Merged-table

This is an object groups multiple customer objects together with a higher capacity and exists out of two or more table objects. the rest of the functionality is the same as a normal table object.

### Bar

This is an object that groups multiple customers that will not be assigned to a table or merged-table.

It has a distinct look compared to the table and merged-table, with a limited size. The size can be determined by the user of the application via radio buttons.

### Waiting area

Values are not changed by interaction of other objects. Is placed by clicking 2 square, the area is then draw as a rectangle between these squares. Customers that are waiting are drawn as dots, depending on the amount of customers waiting. Has a capacity based on the amount of squares

### Group area

Values are not changed by interaction of other objects, always needs one free square.

### Smoking area

Values are not changed by interaction of other objects. Has a capacity based on the amount of squares

### Restaurant plan

The Restaurant plan contains multiple components.

The Restaurant plan is build up as a grid. The maximum of components that can be placed is the equal to the size of the grid. The capacity of customers that can fit in a square is 4.

## Conclusion

The application will have a floor plan that is build up as a grid, this grid can hold as many components as there are squares available.

Each square has a maximum customer capacity of 4.

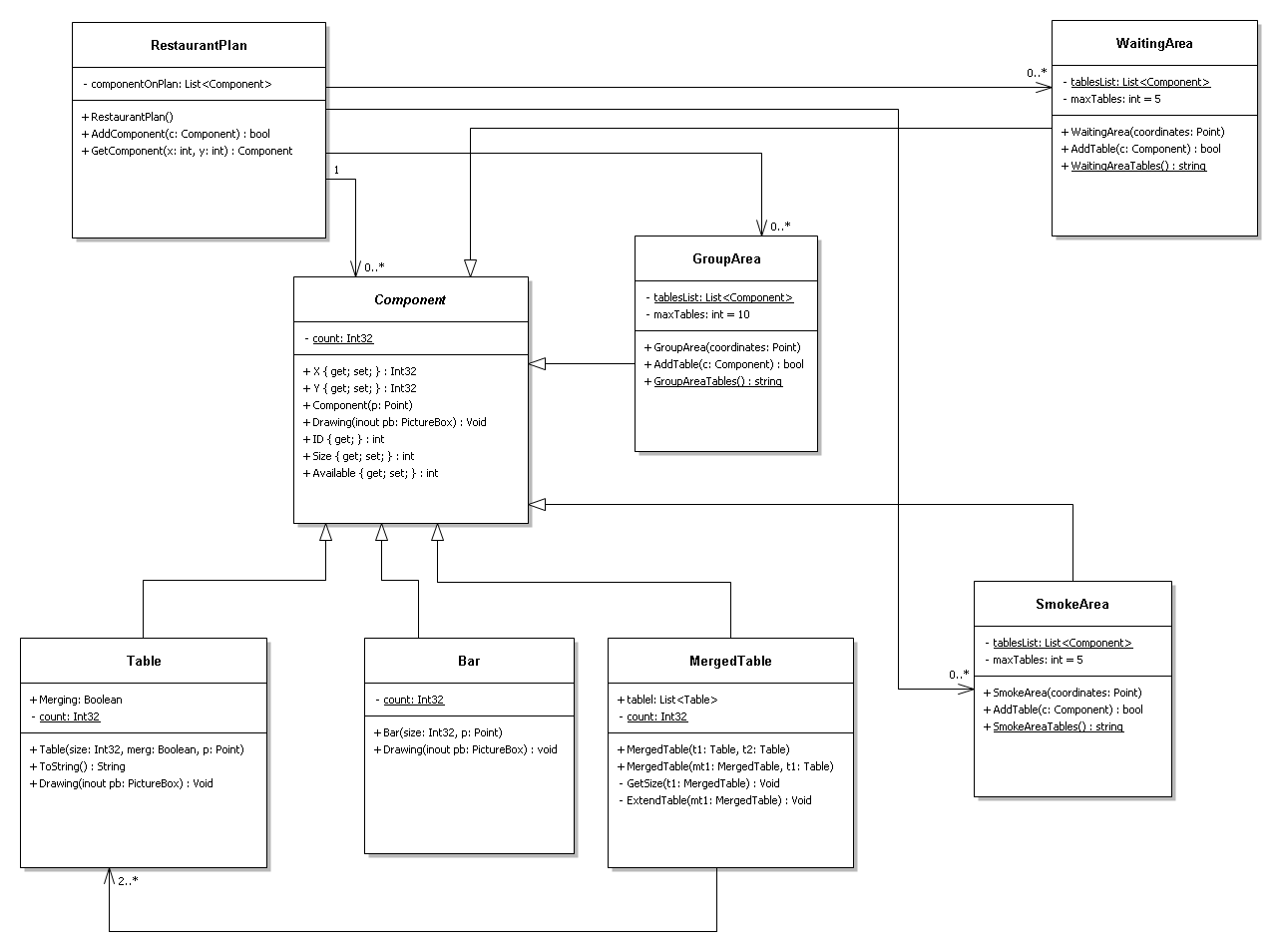
The areas are drawn by clicking on the corner of two different squares, it is then created.

A table can occupy a square and after merging, the merged table has the capability to visually changing size, taking up more squares on the grid.

The bar looks visually different to the table and merged table.

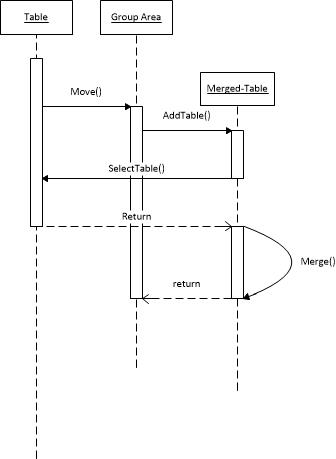
A customer is visualized as a dot on the screen whilst in the waiting area.

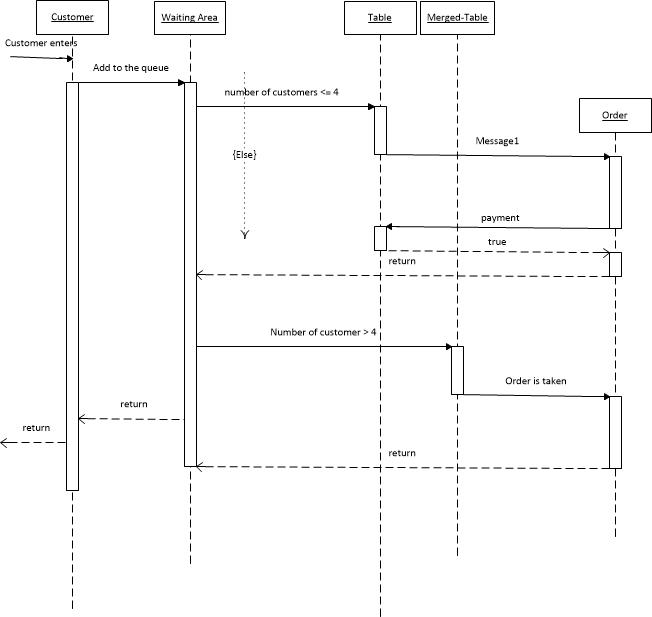
## UML Class diagram



## Sequence diagrams

**Table merging**



****