Capstone Project Report for Phase 1

May 27, 2013

Group #3

Charles Hafslund

Fred Arandi

Mike Utenkov

Othman Smihi

Terry McCarthy

# Table of Contents

|  |  |
| --- | --- |
| Section | Pages |
| Meeting Minutes | 1-6 |
| Individual Contributions | 6-7 |
| Artifacts | 8-22 |
| Vision | 8 |
| Business Case | 8 |
| Actors Goals | 9-10 |
| Domain Model | 11 |
| Main Success Scenario | 12 |
| Use Cases | 13-16 |
| Scope | 17-20 |
| Time Estimates | 21 |
| Risks | 21 |
| Glossary | 22 |

# Meeting Minutes

## Meeting 1 Minutes: Wednesday 5/15/13

### Present: Charles Hafslund, Mikhail Utenkov, Othman Smihi, Terry McCarthy

### Synopsis

Length: approx 2hrs 30mins

The project ideas previously posted on the project ideas wiki page were discussed. After comparing the technical challenges and the project's ability to meet the course goals the Restaurant application was selected. We discussed the domain, actors, goals, and scope of the project. Othman documented the discussion and Terry was going to transcribe the notes electronically. Very briefly we discussed the need for scheduling algorithms and possible fields needed in the ERD. We decided to implement the mobile piece of the application as a mobile website and attack Android development as time allowed. Tools for graphical elements such as the domain model and SD's were discussed, but a clear decision was not made. The group decided on to meet on Tuesday evenings on the off weeks. All group members contributed equally to the discussion. (Synopsis by Charles Hafslund)

### Project Idea

#### A restaurant management application used to…

* Enter, track, and delegate food and drink orders
* Enter and track table statuses
* Handle billing
* Track inventory

### Actors

#### Essential

Waiter

Kitchen staff

#### Important

Hostess

Manager

#### Desirable

Busboys

Bartender

### Entities (incomplete)

Menu

Waiter

Kitchen-staff

Customer

Hostess

Reservation

Tab

Order

Order queue

Dish

Dish queue

Dish prep time

Order prep time (?)

Ingredient

Drink

Bartender

Table

Busboy

Bill

Price

Tip

### Use Case Scenarios (incomplete)

* Wait-staff take orders from customers
* Kitchen staff check off items and coordinate times
* Table management
* Make reservations / order in advance
* Manager reviews/manages inventory
* Record customer complaints
* View item recipe/metadata
* Customer reviews order
* View menu (?)

### Main Success Scenario / General Work Flow

Customer enters

Waiter/Hostess checks for available table

Customer seated by waiter/hostess

Drinks ordered/confirmed/added to tab (?)

Drinks prepared

Drinks delivered

Appetizer dishes ordered/confirmed/added to tab

Appetizers prepared

Appetizers dishes delivered

Order/dishes ordered/ added to tab

Order processed by algorithm

* Orders queued FIFO
* Dishes queued within order by LPF (longest prep-time first) so that all are completed at roughly the same time

Out of scope: check if ingredients are in stock

Cook “checks out” order or dish (badge scan)

Cook ID associated with that order/dish

Cook prepares dish/order

Cook “checks in” order when it is completed (wait staff notified it is ready)

Waiter picks up order and brings it to table, order is marked as “delivered”

People eat food, digest food, excrete remnants (out of scope) ;^P

Loop for other orders

Meal completed / tab closed

Waiter prepares and presents bill

Customers complete payment/tip

Tab designated “settled”

People leave

Table placed in bus queue

Table bussed

Table marked “available”

### Sign-off:

|  |  |
| --- | --- |
| Team Member | Meeting 1 Minutes: Sign off |
| Charles Hafslund |  |
| Mikhail Utenkov |  |
| Othman Smihi |  |
| Terry McCarthy |  |
| Fredrick Arandi |  |

## Meeting 2 Minutes: Wednesday 5/21/13

### Present:

Charles Hafslund, Fred Arandi, Mikhail Utenkov, Othman Smihi, Terry McCarthy

### Othman provided the agenda:

Review/refine whatever "vision" or general statements about the project

Review/refine the actors and entity-relationship diagram.  
Discuss the scope and decide what's in & what's out.  
Settle on some uses cases to flesh out more fully.  
Start putting together the first presentation

Brainstorm a name for our product  
Make a plan for who's working on what until our next meeting.  
If we have time, take a look at using eGit in eclipse. Even though we don't have any code yet, we can use Git to share & maintain whatever artifacts that are more than just text (powerpoints, diagrams, etc).

### Project name

Charles proposed OrDrUp

### Status

Othman started vision statement

Charles went over his story boards and draft of Domain Model diagram.

Fred started work on scope

Mike started use cases

Vision and Scope

What restaurant functions will it be used for?

Mobile version of website.

What restaurants will use this? One restaurant? Most restaurants.

Table layouts?

Basic: list

Medium: xml

High-end: UI

Payment stub

Paying for meal

Make a page for payment

Payment page: Credit info number, expiration, etc.

Email receipt

Pay for app: “Go Pro”

### Actors & Goals

Reviewed actors and goals

<https://github.com/osmihi/Capstone/wiki/Actors-&-Goals>

Charles will update Domain Model to reflect changes to Actors/Goals

### Domain Model

Discussed Charles domain model and made suggestions

Charles white-boarded billing

### Scope

Fred went over scope

<https://github.com/osmihi/Capstone/wiki/Scope>

Group: elaborated what’s IN and OUT and questionable. (Othman at whiteboard)

### Use Cases

Mike went over use case format

Group approved the work done so far

### Project Time Requirements

Othman: How long will the different parts of the project take?

Referred to categories of work: <https://github.com/osmihi/Capstone/wiki/Tools-&-resources>

12 weeks

5 people

3 hours / week / person for meetings

6 hours / week / person outside of meeting

540 hours total

In order of required time (most time first):

|  |  |  |
| --- | --- | --- |
|  | Time-rating | Man-hours |
| App Design | 7 | ? |
| Mobile App | 6 | ? |
| Web app | 5 | ? |
| server code | 4 | ? |
| Testing | 3 | ? |
| Database | 2 | ? |
| Graphics/Design | 1 | ? |

### Power Point Presentation

Standard ppt formatting

Fred: introduction from scope as vision statement and business case

Actors / goals

Domain Model diagram

Highlight main success scenario with story boards

Use Case Table

Risks

App speed

Angry customers

App Reliability

Credit card PCI information handling

Secure logins -> roles

Billing accuracy

Effective prioritization algorithm for orders and dishes

Scope: what’s in / out

Time planning

Cost

Labor

Note: Need to create a glossary for artifacts, see GitHub

### Sign-off:

|  |  |
| --- | --- |
| Team Member | Meeting 2 Minutes: Sign off |
| Charles Hafslund |  |
| Mikhail Utenkov |  |
| Othman Smihi |  |
| Terry McCarthy |  |
| Fredrick Arandi |  |

# Individual Contributions

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Name** | **Description** | **Time (min)** |
| 5/15/2013 | Charles | Initial Meeting | 150 |
| 5/15/2013 | Othman | Initial Meeting | 150 |
| 5/15/2013 | Terry | Initial Meeting | 150 |
| 5/15/2013 | Mike | Initial Meeting | 150 |
| 5/16/2013 | Terry | Meeting notes, organizational communications | 60 |
| 5/19/2013 | Othman | Planning & organization; work on some text artifacts | 90 |
| 5/19/2013 | Mike | Working on use cases. Creating table with priorities,actors, and writing descriptions of cases. | 60 |
| 5/20/2013 | Charles | Sketched out domain model based on actors and entities identified. Created rough draft of domain model in Visio and email it to the group. | 90 |
| 5/20/2013 | Mike | Continued to work on the preliminary use cases. | 60 |
| 5/20/2013 | Fred | drafted scope document | 90 |
| 5/21/2013 | Charles | 2nd team meeting | 180 |
| 5/21/2013 | Othman | 2nd team meeting | 180 |
| 5/21/2013 | Terry | 2nd team meeting | 180 |
| 5/21/2013 | Mike | 2nd team meeting | 180 |
| 5/21/2013 | Fred | 2nd team meeting | 180 |
| 5/22/2013 | Charles | Cleaned up domain model and added elements discussed in the team meeting. Added arrows and relationship descriptions to each element. Posed question to the group on relationships to the Dish object. | 90 |
| 5/22/2013 | Terry | Presentation 1 PowerPoint Presentation | 50 |
| 5/23/2013 | Charles | Reviewed Terry's powerpoint slide deck. Created a couple logo options. | 30 |
| 5/23/2013 | Terry | Presentation 1 PowerPoint Presentation | 60 |
| 5/26/2013 | Othman | Make changes to powerpoint; make additions to wiki; e-mail | 150 |
| 5/27/2013 | Terry | First draft of phase 1 report, group communications | 100 |
| 5/21/1013 | Charles | Started story boarding some of the mobile site and kitchen display for use in the presentation. I also updated a couple relationships in the domain model. | 50 |
| 5/26/2013 | Othman | Make changes to powerpoint; make additions to wiki; e-mail. | 150 |
| 5/27/2013 | Othman | Update powerpoint; make changes to glossary / entities; team communications; add directions for using Git. Also create planning page for future activities | 150 |
| 5/27/2013 | Terry | Compiled first draft of phase 1 report | 60 |
| 5/27/2013 | Fred | Cleaned up scope document and reviewed minutes from previous meeting. | 90 |
| 5/27/2013 | Othman | More email & misc. stuff | 30 |
| 5/27/2013 | Charles | Reviewed PowerPoint and storyboards. Misc emails concerning report and presentation. Brainstorming on Glossary terms. Review of report. Fighting with GitHub. | 100 |
| 5/28/2013 | Charles | Updated PowerPoint. Placed customer interaction with app out of scope. Added basic Idea section, left old slides in that should be removed and changed text color to red. Updated Domain Model. Changed Drink to Beverage. Presentation Planning. | 120 |
| 5/28/2013 | Charles | Created notes for my section of the presentation (DM and Basic idea). Ran through presentation and found I was averaging 5:30. | 45 |
| 5/28/2013 | Terry | Polished report, added data to work log, group communications | 60 |

### Individual Contributions Sign-off:

|  |  |
| --- | --- |
| Team Member | Meeting 2 Minutes: Sign off |
| Charles Hafslund |  |
| Mikhail Utenkov |  |
| Othman Smihi |  |
| Terry McCarthy |  |
| Fredrick Arandi |  |

# Inception Materials/Artifacts

## Vision

### A restaurant management application used to…

* Enter, track, and delegate food and drink orders
* Enter and track table statuses and reservations
* Prioritize orders in the kitchen according to arrival of orders and required prep times
* Handle billing
* Track inventory

### We envision this as a mobile web application which may be adapted to Android if time allows.

## Business Case

##### Restaurants are busy environments and inefficiencies can result in business failure even if the food is great.

##### Effective coordination of personnel and resources is key to timely service and customer satisfaction.

##### Boost business by

* Making waiting times smaller and more predictable
* Maximizing customer throughput
* Maximizing employee efficiency
* Increasing customer satisfaction

## Actors/Goals

### Waiter

* Enter customers' orders quickly and easily
* Enter customers' orders without errors
* Communicate customer orders to the kitchen
* Access menu information on demand, as needed
* Adjust bill splitting/sharing as needed
* Mark what is from each person (seat number)
* Take customer payment quickly and easily
* See specials
* Use coupons / codes
* Receive tips
* Take complaints

### Kitchen staff

* View clear and accurate report of queued orders
* View upcoming orders in terms of priority
* Take ownership of orders
* Mark items as completed

### Customer

* Have order recorded accurately
* Receive an accurate bill, including any splitting/sharing arrangements
* Pay with a credit card easily and securely
* Use coupons / codes
* Make a reservation
* Lodge a complaint

### Hostess

* View which tables are open and which are not
* View waiting times for tables via check in and check out, average duration
* View which tables may be open soon
* See table capacities
* View table server assignments by zone

### Busboy

* Indicate that tables are ready for seating, easily and without error
* See which tables need to be cleaned (indicated by bill paid)

### Bartender

* View queue of drinks to be made
* Mark drinks as served

### Manager

* View aggregated information about sales
* View aggregated information about inventory
* View aggregated information about employee work
* Make adjustments to bill amounts
* View complaints
* Perform all other roles

## Domain Model

## Main Success Scenario

This is the general series of events for the usage of the system, across the various actors, to complete a single customer's experience from start to finish.

*note: Manager's goals are not represented.*

1. Customer enters
2. Waiter/Hostess checks for available table
3. Customer seated by waiter/hostess
4. Drinks ordered/confirmed/added to tab (?)
5. Drinks prepared
6. Drinks delivered
7. Appetizer dishes ordered/confirmed/added to tab
8. Appetizers prepared
9. Appetizers dishes delivered
10. Order/dishes ordered/ added to tab
11. Order processed by algorithm

* Orders queued FIFO
* Dishes/items queued within order by LPF (longest prep-time first) so that all are completed at roughly the same time

1. Out of scope: check if ingredients are in stock
2. Cook “checks out” order or dish (badge scan)
3. Cook ID associated with that order/dish
4. Cook prepares dish/order
5. Cook “checks in” order when it is completed (wait staff notified it is ready)
6. Waiter picks up order and brings it to table, order is marked as “delivered”
7. People eat food, digest food, excrete remnants (out of scope) ;^P
8. Loop for other orders
9. Meal completed / tab closed
10. Waiter prepares and presents bill
11. Customers complete payment/tip
12. Tab designated “settled”
13. People leave
14. Table placed in bus queue
15. Table bussed
16. Table marked “available”

## Use Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Name** | **Scope** | **Primary Actor** | **Complexity** | **Priority** |
| 1 | Menu view | In | Anyone | Low | 1 |
| 2 | Order placement | In | Customer | Low | 1 |
| 3 | Order acquirement | In | Waiter | Low | 1 |
| 4 | Order review | In | Waiter/customer | Low | 1 |
| 5 | Table management | In | Waiter | High | 2 |
| 6 | Order check off | In | Kitchen staff | Medium | 1 |
| 7 | Reservation | In | Customer | Medium | 2 |
| 8 | Item's metadata | In | Anyone | Medium | 2 |
| 9 | Inventory | In | Manager | Low | 1 |
| 10 | Complaints record | In | Manager/waiter | Medium | 2 |

#### Use case #1

**Name:** Menu View.

**Summary:** Customer reviews menu for order placement. Manager checks menu for revision. Waiter reviews menu for knowledge improvement.

**Actors**: Anyone.

**Description:**

1. A customer checks the menu for consecutive order placement, ordering entrée, main course, desert and drinks.
2. Waiter reviews menu for the better understanding of available selections in case if suggestion is asked, or for promotion of specials.
3. Manager reviews menu for revision to exclude less popular items and/or to add additional ones.

**Postcondition:** Customer is ready to place an order, menu is revised and improved.

#### Use case #2

**Name: Order placement.**

**Summary:** Customer reviewed menu and ready to place an order.

**Actors:** Customer.

**Description:**

1. Customer selects an entrée.
2. Customer selects main course.
3. Customer orders a drink.
4. Customer orders desert.

**Exception:** If ordered drink contains alcohol, a form of age identification is used. If customer below 18, drink is rejected and alternative choice is given.

**Postcondition:** Order is placed.

#### Use case #3

**Name: Order acquirement**

**Summary:** Waiter takes an order from every party member at the table.

**Actors:** Waiter.

**Description:**

1. If specials are available, those are promoted first.
2. Waiter takes order of each party member at the table.
3. If variety of side dishes is available, customer is being asked to make selection.
4. Waiter takes order for drinks.
5. Waiter takes order for desert.

**Exception:** If dish is not available, substitution is proposed. If customer doesn’t like what he is being suggested, provide menu for selection. If ordered drinks contain alcohol perform age verification. If customer is below 18, politely reject his choice and ask to make another selection.

**Postcondition:** Orders are taken.

#### Use case #4

**Name: Order review**

**Summary:** Waiter repeats orders back to party members at the table to make sure those are correct.

**Actors:** Waiter/Customer

**Description:**

1. Waiter repeats orders back to each customer.

**Exception:** If order is incorrect, customer corrects waiter.

**Postcondition:** Orders are verified and ready to be relayed to the kitchen.

#### Use case #5

**Name: Table management**

**Summary:** Table is being cleaned after customers have left. If customers are at the table, ask for refill/ or desert. If customers are ready to check out, serve the bill/collect payments. If complaint is filed, provide information to manager on duty.

**Actors:** Waiter, Manager.

**Description:**

1. If being called, approach table and tend to customer’s need.
2. If customers have left, collect dishes and clean table.
3. If customers are present, ask if refill is needed, or if deserts are desired.
4. If deserts are ordered, refer to Use case#3.
5. If customers are ready to pay, provide with bill, collect payments and provide receipts.
6. If complaint is filed, relay information to manager for further assistance, Use case#10

**Exception:** If refill is asked and drink contains alcohol, advise that additional fee will be charged.

**Postcondition:** If table was being cleaned, mark table as vacant. In case of complaint, manager approaches table and manages it.

#### Use case #6

**Name: Order check off**

**Summary:** Kitchen staff checks of the order after orders review

**Actors:** Kitchen staff, Waiter

**Description:**

1. Kitchen staff reviews order.
2. Kitchen staff checks off orders.
3. If certain ingredients for selected dish are not in stock, exception is thrown.
4. Kitchen staff begins to work on order.

**Exception:** If certain ingredients are missing for selected dish, page waiter and advise him to notify customer and retake the order.

**Postcondition:** Order’s preparation is started.

#### Use case #7

**Name: Reservation**

**Summary:** Customer calls to make a table reservation.

**Actors:** Customer, waiter/host.

**Description:**

1. Customer calls in to make a reservation.
2. Waiter/host checks reservation records for availability.
3. If table is available, reservation is accepted, otherwise exception is thrown.

**Exception:** If all table are occupied or number of reservations exceeds number of available tables notify customer about expected waiting time.

**Postcondition:** Reservation is being taken.

#### Use case #8

**Name: Dish’s description**

**Summary:** A throughout description of available dishes is available on the menu.

**Actors:** Anyone.

**Description:**

1. Customer can review ingredients for Medical purposes or preferences.
2. Waiter can review ingredients for better understating of the offered dishes and to notify customers that certain products might contain certain products that can cause allergic reaction.
3. Manager can review ingredients for inventory and product’s order placement.

**Postcondition: N/A**

#### Use case #9

**Name: Inventory**

**Summary:** Manager performs inventory to keep track of available products, for documentation purposes and to place an order for items that are low in stock.

**Actors:** Manager.

**Description:**

1. Manager checks available products.
2. If certain product is 0 quantity, place an order for it and throw an exception.
3. If certain product is low on quantity, place an order throw an exception.
4. Document items that has been used, left of in high demand.

**Exception:** If product is 0 quantity, manager places order and notifies waiters not to take orders for dish containing that product until further notice. If product is low on quantity, place an order on it and advised waiters to check quantity of mentioned above product ,if order is taken and contains dish from that product, to avoid disappointing customers.

**Postcondition:** Inventory is completed, and orders are placed.

#### Use case #10

**Name: Complaint management**

**Summary:** Customer files a complaint, which is needed to be settled with.

**Actors:** Manager, Waiter, Customer.

**Description:**

1. Waiter approaches manager and notifies him about complaint being filed.
2. Manager approaches complainer’s table and attempts to resolve the situation.

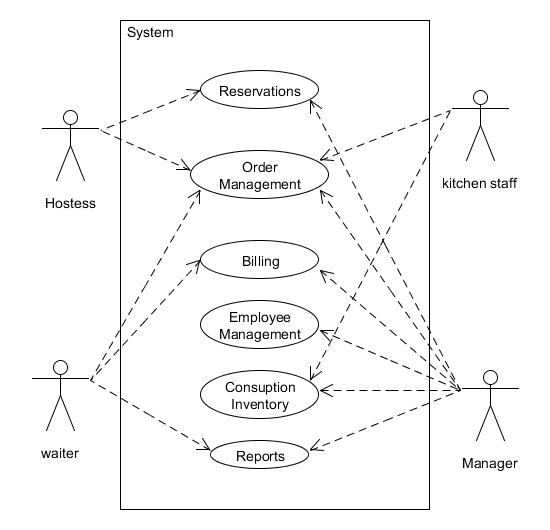
**Postcondition:** Situation is resolved.

## Scope

OrDrUp will:

* Accurately enter track and delegate customer food and drink orders.
* Monitor and keep track of reservations and availability status in real time.
* Prioritize orders in the kitchen according to arrival of orders and required prep times
* Handle billing and promotional offers
* Track inventory
* Manage employees access

The overall scope of the project is reflected in the use case diagram shown below



Reservation management include

~~Adding deleting and confirming reservations~~

Finding open tables

Assigning open tables

Capacity

Customer time/ in out

Drinks

##### Order management include

Adding, changing, deleting and closing orders

Dish classification/categories

Accessing menu items

Relay customer order to kitchen staff

Kitchen UI(queue display)

##### Billing services will cover

Cash, credit and gift card processing.

Bill splitting

Bill adjustments

Tips

##### Employee management includes

Scheduling, adding and removing employees from database.

Basic account security – login/ roles

##### Manager should be able to

View aggregated information about sales report

View aggregated information about consumption report

View aggregated information about employee work

Make adjustments to bill amounts

Comping/discounts

##### Inventory management will

~~Show current stock~~

show consumption invemntory

Give notice when stock goes below a specific level.

##### Introduction / Background to project

Restaurants are very busy and if they are inefficiently run can fail even though they are serving great food. Without an application that can keep track of average wait times and the lengths of time spent dinning by customers, it is hard to predict how long the wait time will be for new customers and as we all know customers are very impatient. This application should track and provide the user with data relating to customer wait times, reservation and seating, accounting and auditing, staffing and real time consumption inventory. Leaving the staff to what it should be doing, good food and excellent dining experience

##### Business case

Boost business by

* Making waiting times smaller and more predictable
* Maximizing customer throughput
* Maximizing employee efficiency
* Increasing customer satisfaction

##### Key people / Key stakeholders

**Project-team members**

Charles Hafslund,

Mikhail Utenkov,

Othman Smihi,

Terry McCarthy,

Fredrick Arandi

**Client** – Order up Restaurant

##### In scope

* Queue customers waiting to be seated
* Table management:
  + Table capacity
  + Server assignments
  + Ready to clean / cleaned status
* Menu
* Drinks as items
* Dish categories (app, main, dessert, drink, etc.)
* Kitchen UI (queue display)
* Send orders to kitchen
* Prioritize orders in the kitchen
* Mobile site
* Bill splitting
* Payment API (stub)
* Tips
* Customer time in/out
* Busboy role
* Basic account authorization & security
* Employee names and roles
* Consumption report of ingredients
* Sales numbers

##### Out of scope

* Parties exceeding 1 table
* Specific of payment/PCI
* Payroll/ time
* Inventory management
* Complaints
* Bar management
* No sharing

##### Project administration, monitoring and reporting

Project meetings shall be held every Tuesday in person or via Skype, whichever agreed upon in the previous meeting

Reports and updates shall be logged in github. - Members have to sign documents at each meet

## Time estimates

### 12 weeks

### 5 people

### 9 hours / person-week

### 540 hours total

|  |  |
| --- | --- |
| Project Aspect | Man-hours |
| App Design | 140 |
| Mobile App | 120 |
| Web app | 120 |
| Server code | 60 |
| Testing | 50 |
| Database | 25 |
| Graphics/Design | 25 |

## Risks

* **Fast application speed: Do not try customers’ patience.**
* **High reliability: Downtime must be avoided**
* **Credit card PCI information handling**
* **Billing accuracy**
* **Logins securely connect users with their correct role**
* **Effective prioritization algorithm for orders and dishes**

## Current glossary of terms

Bill: All Dishes belonging to an Order Group. Includes the price of each Dish and the total price

Bus queue: List of tables ready to be cleaned and prepared for new customers

Discount: Any action that reduces prices below those listed in the menu. Includes coupons, complimentary items, manager adjustments

Item Type: An attribute of a menu item; values include the basic categories found within the menu: Appetizer, Main, Dessert, Drink (non-alcoholic), Drink (alcoholic), etc.

Menu Item: A dish or drink on the menu that a customer can order

Item Queue: List of Items, used in the kitchen to plan for Items of the same Type to be ready for each seat at a table at the same time

Order Group: Orders to be paid by the same payee

Order Queue: List of Orders, used in the kitchen to plan for orders to be prepared on a first-come first-served basis (grouped by Party and Dish Type)

Order: A set of items ordered by a customer at a seat

Party: A set of customers sitting together and ordering for the same meal

Seat: A single spot at a Table, identified by number. A Customer in a Party occupies a Seat.

Service Queue: Collection of Items of the same type ready to be delivered to a table

Table status: Attribute of Tables determining next action that can be taken. Values include: occupied, bus-queue, available

Table: The physical location at which Customers sit while dining