[Date]

TableSmart: Vision Document

Yifang Wang
Logan Medford
Lou Alexander Salvador
Yesol Lee
SOFTWARE METHODOLOGIES SECTION 4

Revision History

DATE	VERSION	DESCRIPTION
01/25/2017	0.1	1st Draft
01/30/2017	0.2	2nd Draft
02/02/2017	0.3	3rd Draft
02/06/2017	0.4	4th Draft
02/15/2017	0.5	5th Draft
02/16/2017	0.6	Final Part A
03/22/2017	0.7	Part B
03/23/2017	0.8	B 1.1

Contents

Problem Description

Working in a restaurant can be very demanding depending on location and business traffic – having to deal with all the reservations and table seating can be very hectic and chaotic if left unsupervised. Hosts and hostesses are responsible for monitoring reservations, tables and seating allocation within the restaurant. SIR Corp's Jack Astor's is a widely popular restaurant, often having a full house especially throughout patio season. However, Jack's is currently facing a problem due to a primitive reservation system and a lack of dependable table tracking system implementation. Due to lack of proper communication channels, table wait times have been an ongoing problem, affecting the overall customer satisfaction of the guests, as well as adding stress to the front of house team.

It is recommended that a new system be developed and deployed make reservations and table tracking an easier and smoother for both guests and staff alike. This system should be deployed on portable devices such as tablets and smartphones.

System Capabilities

The new system is capable of:

- Real-time table and section status tracking.
- Table addition/merge option for parties 6 or more.
- Clear table option after resetting a table.
- Reservation record input based on phoned reservations.
- Alert host of booked reservation 2 hours prior to event.
- Table queue tracking list
    - Add/Remove guest name
    - Set queue timer to ensure that wait time is of acceptable length.
- Storing and generating statistic reports on wait time and table turnovers.

Business Benefits

It is anticipated that deploying this new system will provide the following benefits for the

restaurant:

- More organized tables and sections thereby providing smoother transition between tables.
- More communication between Front of House and Back of House teams thereby elevating overall guest experience.
- Reservation system integration creates a more structured work flow for hosts and managers thereby greatly reducing work environment stress.

Business Related Questions

No.	*Stake-holder*	*Question*	*Answer*
1.	Owner	How much is you budget?	$30,000
2.	Owner	How long you want this project to be done?	3 months.
3.0		*Reservation error:*	
3.1	FoH manager	We will associate with the reservation system, but what if there isn't a reservation in the system?	The host should call the manager immediately, and manager will make decision there's nothing the system would help.
3.2	FoH manager	Do you need a recording feature for these errors?	Yes, a small one, to keep issued time, manager's name, and order list number. It should be done in one or two steps (staff card tap or click & staff number received). This only for matching the detailed record in order system. It doesn't need to contain more features.
4.0		*Queue list*	

4.1	Host	Do you want the queue list part on the interface? Or you want it show up only when you click the feature button?	I prefer a foldable GUI for this feature, so we can keep it with tables view when needed, and fold it when it's not useful.
4.2	Host	Do you want the queue only show the first name for each table size?	Yes. Sort by "table for 2", "table for 4", and "table for more (with specific numbers)".
4.3	Host	When you create a name to the list, do you want to organize it (to appropriate list) automatically or manually?	Automatically.
4.4	Host	Do you want a number present how many names under this table size on the interface? Like "Queue of table for 2 (XX waiting)".	Yes.
5.0		*Merge table*	
5.1	FoH manager & Host	Do you have a preference for how to merge tables? A button around table picture or a feature when you click a table to assign?	The second one, click a table then it will bump up options "assign", "merge", and "unmerge" to in case guests want to move.
5.2	FoH manager & Host	How do you want to unmerge a table after the guest left? Do you want host to do that or the busser?	It should automatically unmerge after busser reset table to system.
6.	Owner & Managers	What device do you wish to implement the system on?	IPad and/or tablet
7.	Bussers & Servers	Do you wish to integrate table availability system with smartphones?	Yes, for easy access and real-time update.

Activity Diagrams

Workflow List:

- Table Merge/Addition
- **Table Availability**
- **Guest Seating**
- Reservation Assignment
- **Table Queueing System**
- Report Ad hoc System

We have selected to provide an activity and workflow diagram to the following three: Guest Seating, Table Availability and Table Queueing systems.

Guest Seating

Description

The guest seating system involves the host or manager to interact with guests taking in information in regards to reservations, party number and seating preferences which are vital in assessing and assigning table arrangements for guests.

1.0 Guest Seating Activity Diagram

1.1 Acquire table-related information. (e.g. how many people, section preference)

1.2 Check system for open section and available tables.

1.2.1 If the party is greater than 6:

1.2.1.1 Add/Merge tables

1.2.2 If there are no available tables

1.2.2.1 Add guest name on queue list until a table is free

1.3 Assign table to guest party

1.4 Lead/Send party to table

1.5 Go back to station

Guest Seating Workflow Diagram:

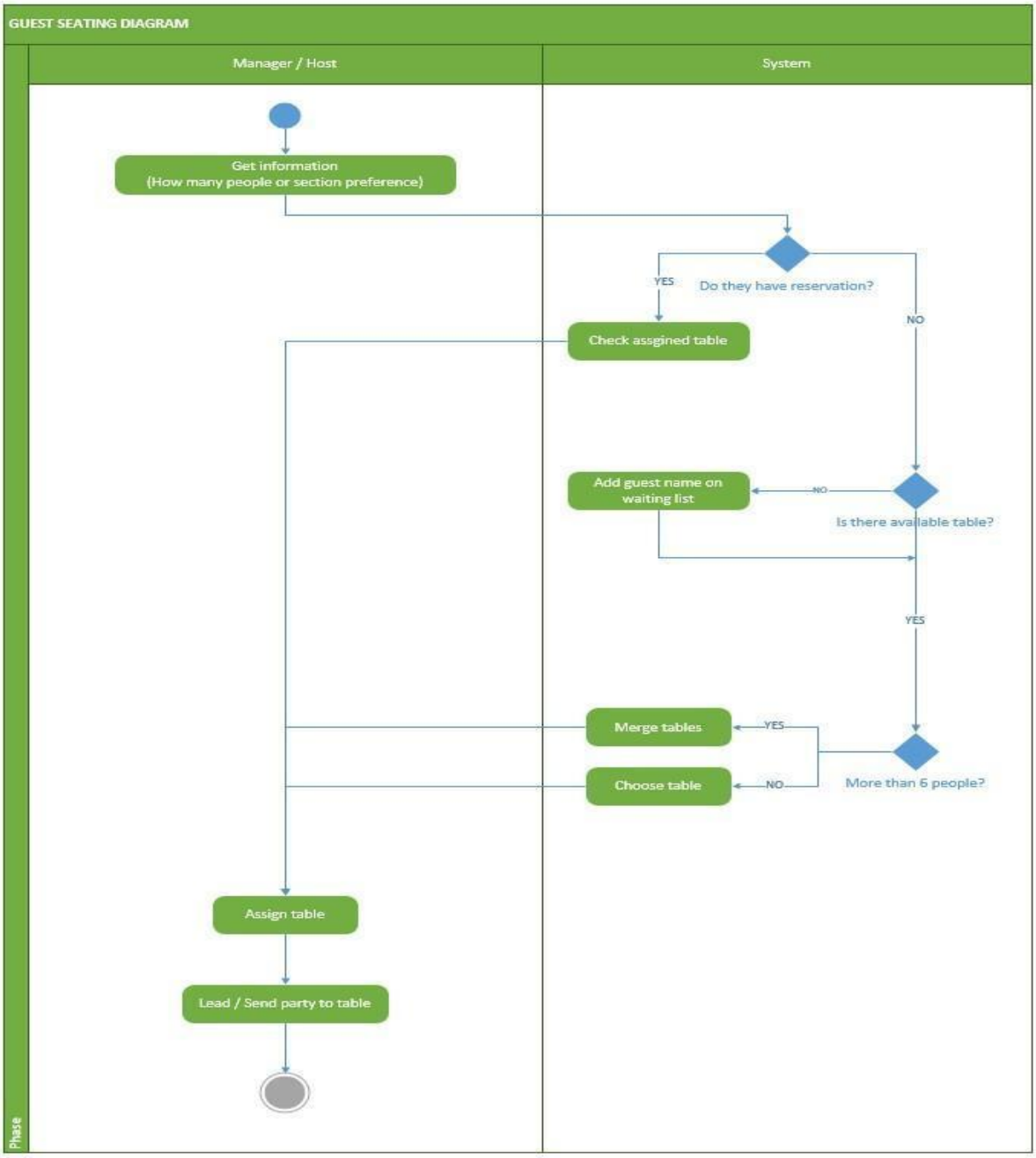


Table Availability System

Description

The table availability system is used for table resetting. This is when a busser clears up and resets a table which goes from occupied to available in the TableSmart system.

2.0 Table Availability Activity Diagram

2.1 Clear everything on table

2.2 Prepare table

2.3 Choose ‘clear table’ option

2.4 System prompts and validates table availability

2.5 Busser confirms with system

2.6 System changes status from occupied to available

Table Availability Workflow Diagram:

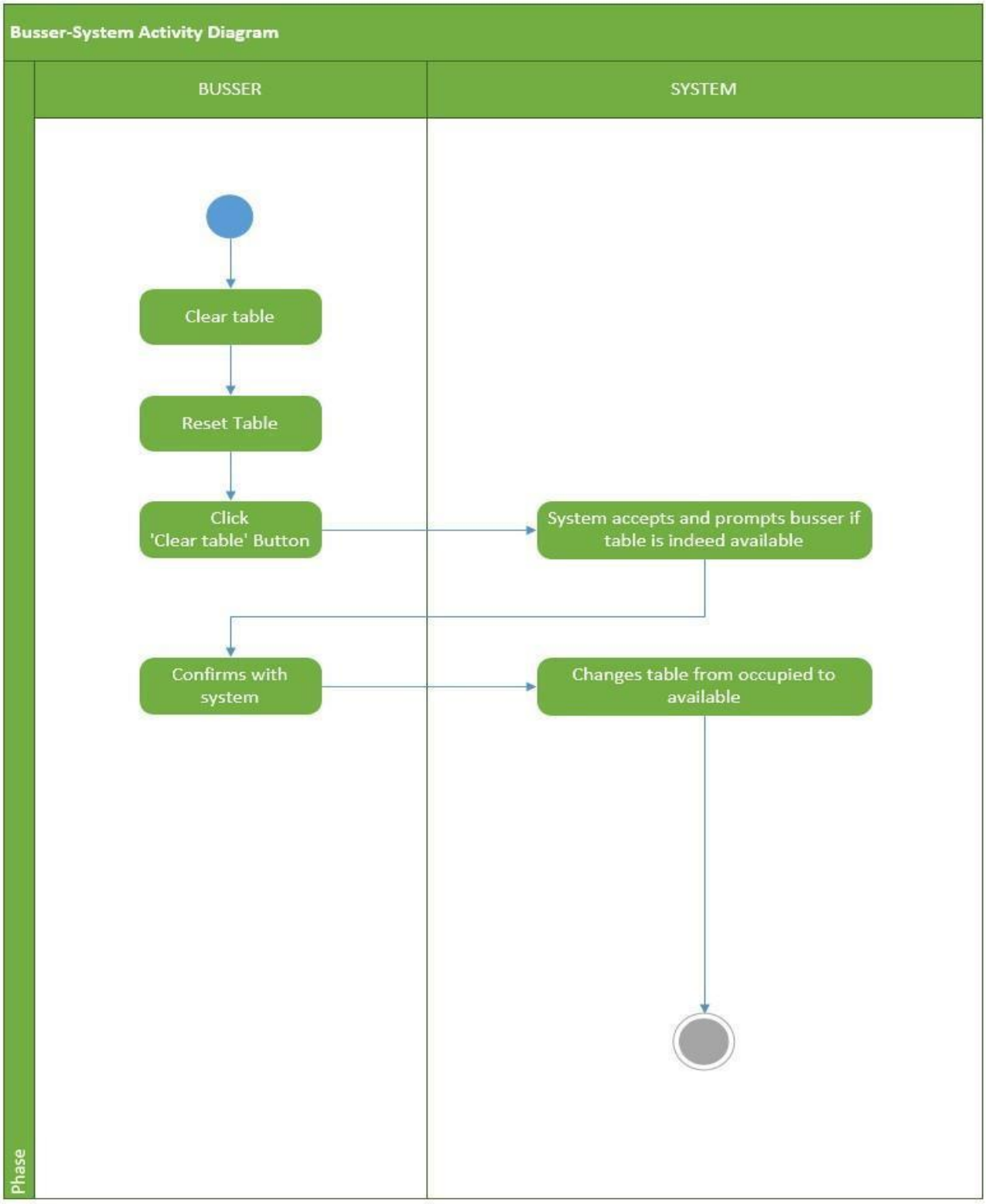


Table Queueing System

Description

This system is responsible for creating and keeping track of guests in queue for the next available table

3.0 Table Queueing System Activity Diagram

3.1 Host/Manager takes in name, party size and section preference

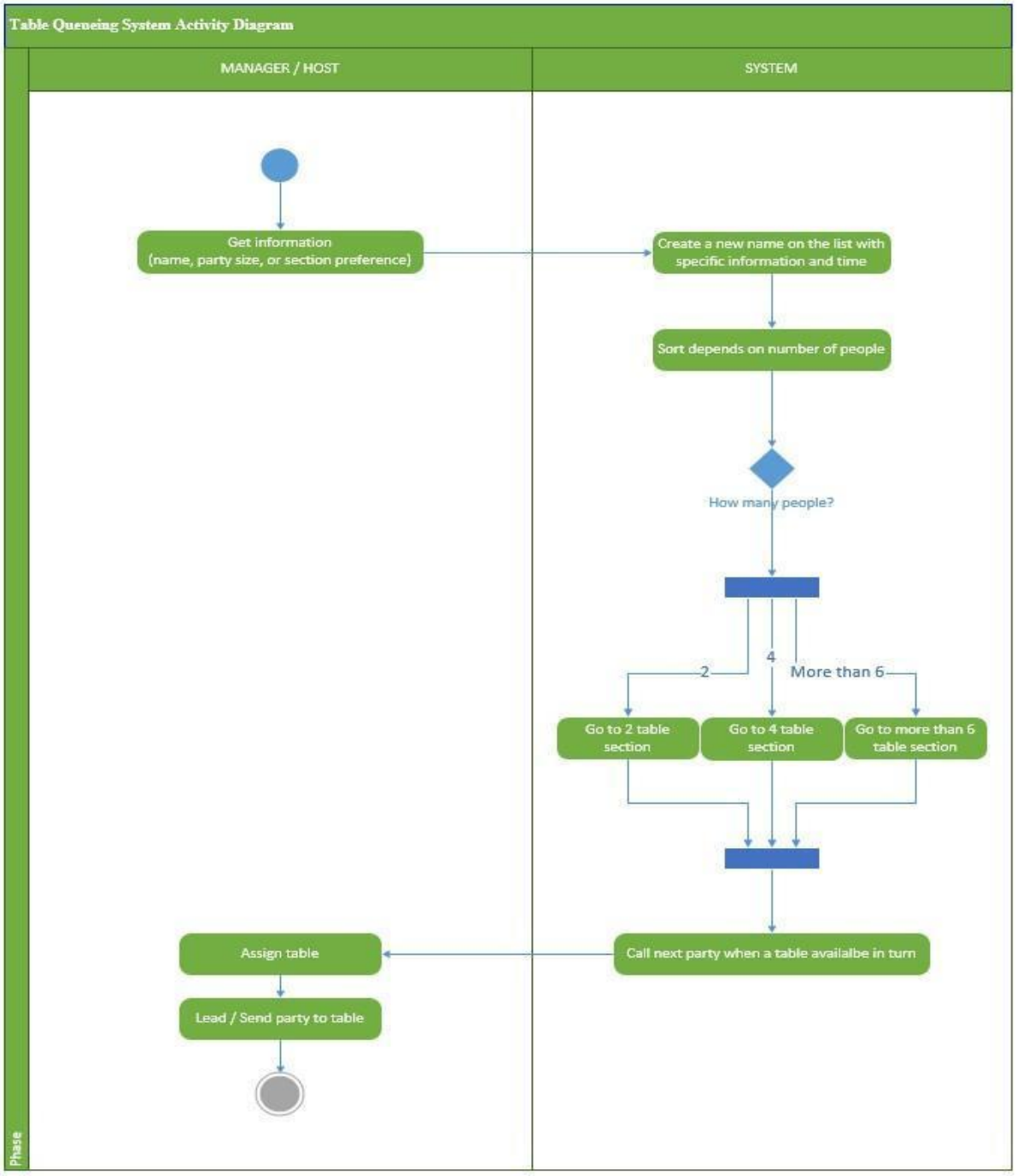
3.1.1 Create a new name on the list with specific information and time

3.2 System records name and party size

3.3 System then sorts the party under the appropriate wait list. (e.g Table for 2, 4, or more)

3.4 System notifies host/manager of the next person in queue

Table Queueing System Workflow Diagram



Use Case Model

Table 1: Actor-Goal Use Case *

NO	USERS' ROLES/ACTORS	GOAL USE CASES
1	Host/FoH Manager	Add reservation Modify reservation Cancel reservation Check available tables Add table waiting list Modify table waiting list Delete table waiting list Assign table Merge table Clear table
2	Busser	Reset table

Table 2: Goal Use Case Description

NO	GOAL USE CASES	USE CASE DESCRIPTIONS
1	Add reservation	Host/FoH manager receive reservation request from guest (phone call or in person) and create a guest record into reservation system: reservation time, group size, seating preference, guest name, phone number, secondary phone number(optional); system confirms reservation.
2	Modify reservation	Host/FoH manager search reservation record with guest name and/or phone number and modify the reservation information; system confirms and updates the reservation record.
3	Cancel reservation	Host/FoH manager search reservation record with guest name and/or phone number; system confirms and delete the reservation record.

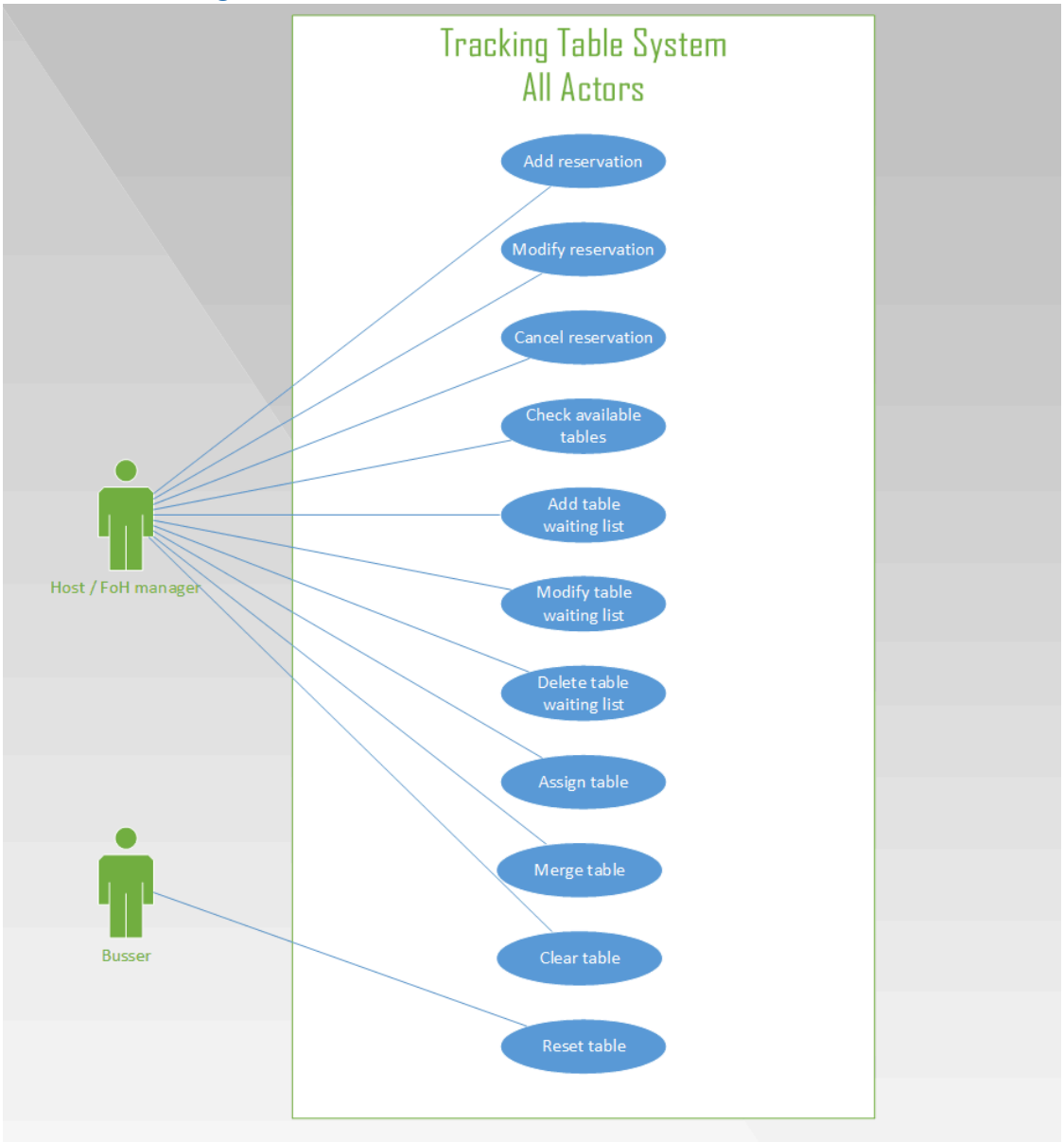
4	Check available tables	Host/FoH manager checks for available table in system based on the guest's information (group size, seating preference); system returns a list of appropriate available tables, separated into two sections: preferred and generic.
5	Add table waiting list	Host/FoH manager add guest into appropriate waiting list with name, group size, and phone number. **Assuming guest don't have any seating preference when they are waiting for a table.*
6	Modify table waiting list	Host/FoH manager select list item with guest name and/or phone number then modify the waiting list information (name, phone number, group size) ; system confirms and updates the record; if group size changed, assign it to appropriate list based on its current size and ticket issued time.
7	Delete table waiting list	Host/FoH manager select list item with guest name and/or phone number then deletes guest record from the waiting list.
8	Merge table	Host/FoH manager combines two or more available tables as one table on the system.
9	Assign table	Host/FoH manager assigns an available table to a guest. If table assigned with reservation or waiting record, remove that record from system.
10	Clear table	Host/FoH manager remove current table information and merge data; only available before order input.
11	Reset table	Busser changes table status from table occupied to available, merge data automatically removed.

Table 3

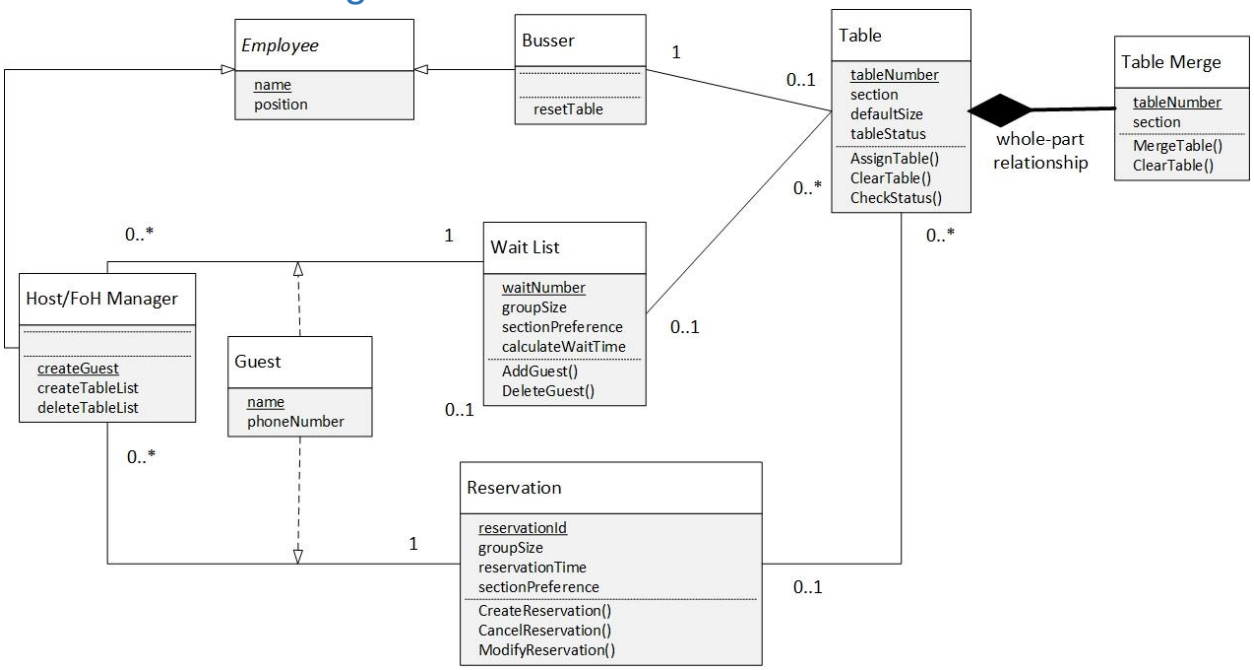
-- One for each sub system/Module followed by a Use Case Diagram

NO	GOAL USE CASE	USERS ROLE/ACTORS
1	Add reservation	Host/FoH manager
2	Modify reservation	Host/FoH manager
3	Cancel reservation	Host/FoH manager
4	Check available table	Host/FoH manager
5	Add table waiting list	Host/FoH manager
6	Modify table waiting list	Host/FoH manager
7	Delete table waiting list	Host/FoH manager
8	Assign table	Host/FoH manager
9	Merge table	Host/FoH manager
10	Clear table	Host/FoH manager
11	Reset table	Busser

Use Case Diagram



Domain Class Diagram



Multiplicity Descriptions:

For one Host there is 0 to many Guest.

There needs to be one Guest to 0 to many Reservation.

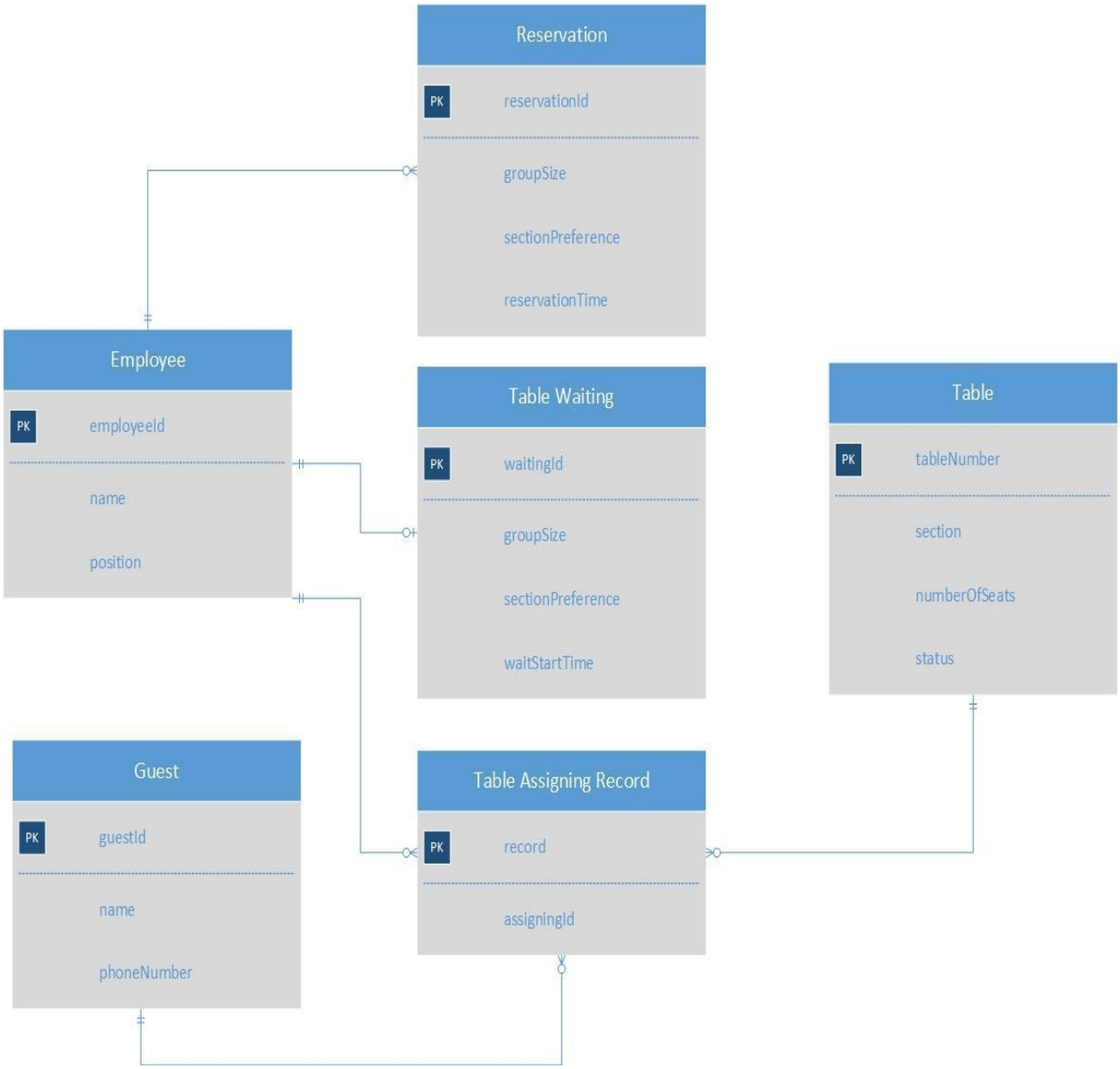
There can be up to 1 Reservations for zero or more Tables

One Busser can have up to one table.

There can be up to one waitlist for multiple tables.

There can be up to one Guest on a wait List.

ERD


Reservation
PK
reservationId
groupSize
sectionPreference
reservationTime
Employee
PK
employeeId
name
position
Table Waiting
PK
waitingId
groupSize
sectionPreference
waitStartTime
Table
PK
tableNumber
section
numberOfSeats
status
Guest
PK
guestId
name
phoneNumber
Table Assigning Record
PK
record
assigningId

Fully Developed Users' Story

Add reservation

Use case name:	*Add reservation*
Scenario:	Add reservation with guest information on system
Triggering event:	Guest requests reservation
Brief description:	Host/FoH Manager put guest's information such as name, contact number, party size, available time…and so on
Actors:	Host / FoH Manager
Related use cases:	Modify reservation, Cancel reservation
Stakeholders:	Host / FoH Manager
Preconditions:	Request reservation from guest must be available
Postconditions:	Guest information (name, contact number, available time etc) must be created and saved information must be associated with Guest.
Flow of activities:	Actor / System 1. Host / FoH Manager turns on reservation system 1.1 System creates a new reservation 1.2 System prompts for guest name and contact number 2. Host / FoH Manager puts guest's name and contact number 2.1 System writes guest's name and contact number in reservation 2.2 System prompts for party size and available time 3. Host / FoH Manager puts guest's party size and available time. 3.1 System writes party size and available time in reservation 3.2 System makes the guest's reservation.
Exception conditions:	2.1 Guest doesn't have contact number 3.1 All table is full at the time

Add table waiting list

Use case name: *Add table waiting list*

Scenario:	Add guests in table waiting list when table is full
Triggering event:	After guest request table, when there is no available table
Brief description:	If there is no table, enter waiting list system. Add guest's information (name, contact, and party size)
Actors:	Host / FoH Manager
Related use cases:	Request table and Check available tables
Stakeholders:	Host / FoH Manager
Preconditions:	• Guest must exist • Request table must be available • Available tables must no exist
Postconditions:	• Guest information (name, contact number, party size) must be created and saved • Waiting list is created and associated with table • information must be associated with Guest.
Flow of activities:	Actor / System 1. Guest request tables 1.1 System looks up available table 1.2 If there is no space, system prompts for guest name and contact number and party size 2. Guest gives name contact, and party size 2.1 System create waiting list with guest's information 3. Check available table 3.1 If there is available table, call guest or text to guest 3.2 Remove waiting list
Exception conditions:	1.1 System can accept waiting list before restaurant finish an hour ago 1.3 All table is full

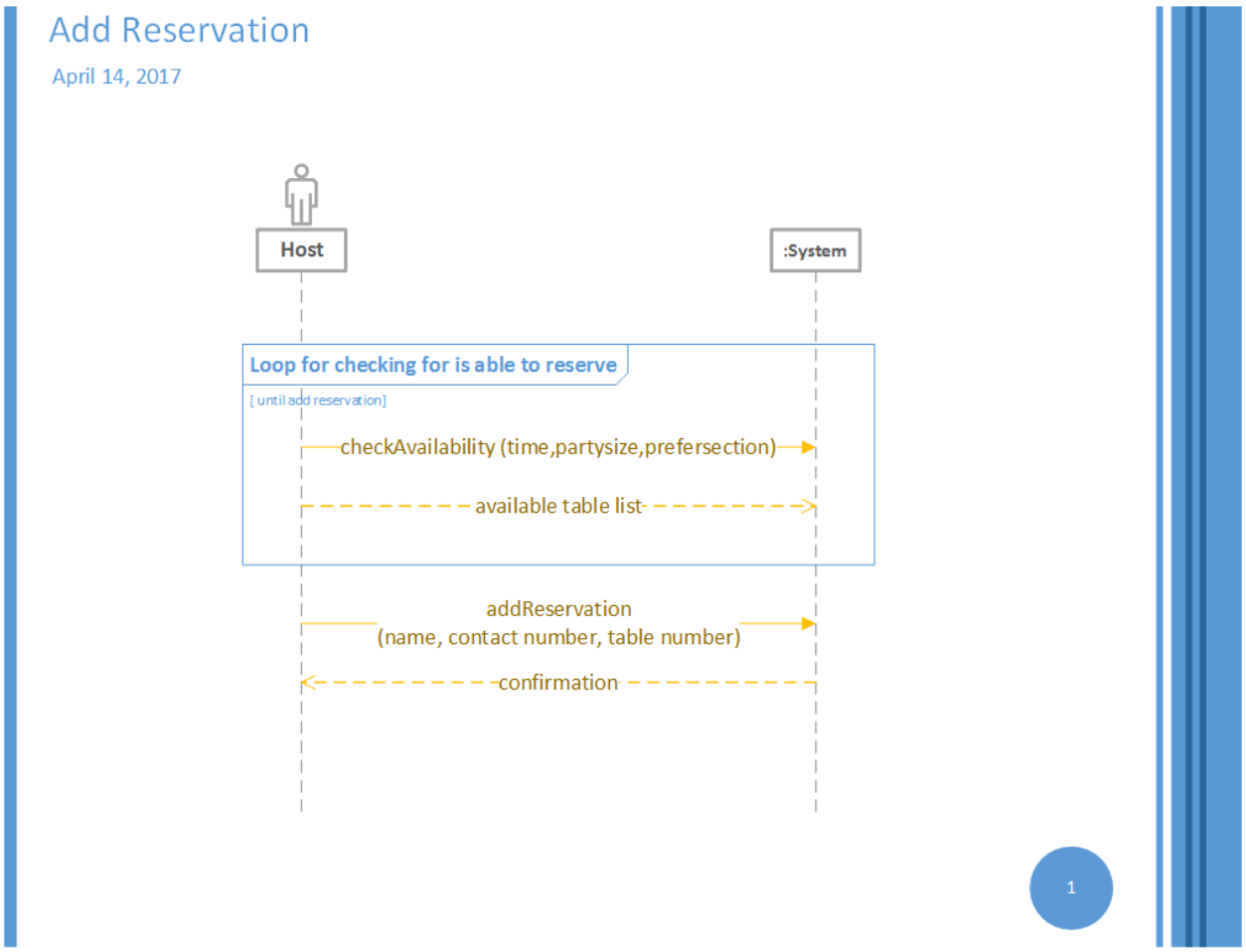
Assign table

Use case name: *Assign table*

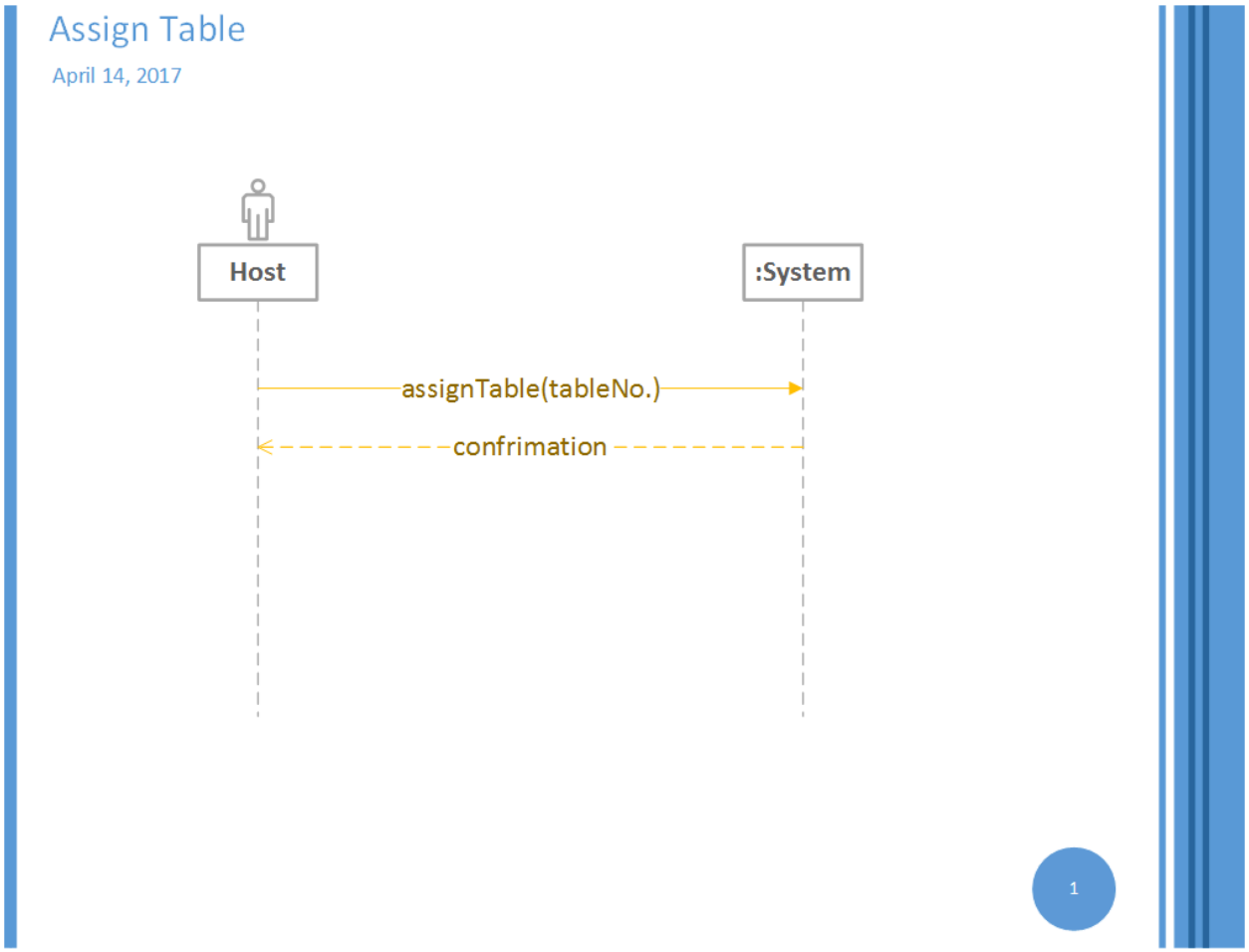
Scenario:	Assign table with customer information
Triggering event:	After guest request table, when there is available table
Brief description:	If available table, enter table system. Add guest's information (name, contact, and party size)
Actors:	Host / FoH Manager
Related use cases:	Request table and Check available tables
Stakeholders:	Host / FoH Manager
Preconditions:	• Guest must exist • Request table must be available • Available tables must exist
Postconditions:	• Guest information (name, contact number, party size) must be created and saved • Table created and associated with guest • information must be associated with guest.
Flow of activities:	Actor / System 1. Guest request tables with the name 1.1 System looks up available table 1.2 If there is space, system prompts table system 2. Host / FoH Manager assign table 2.1 System create table with guest's name
Exception conditions:	1.1 System can accept add table before restaurant finish 30 minutes ago 1.3 Table is available

System Sequence Diagram

Add Reservation

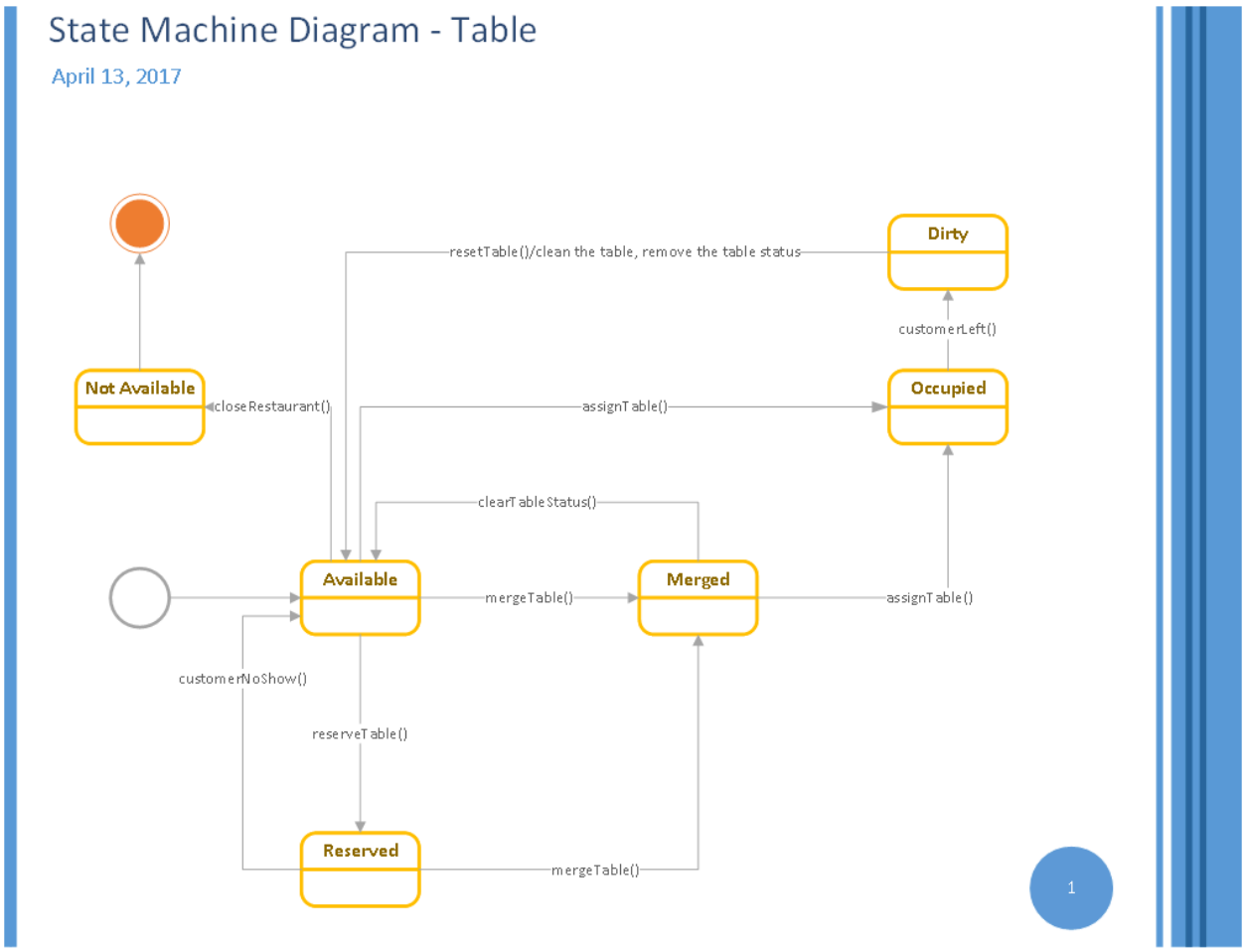


Assign Table

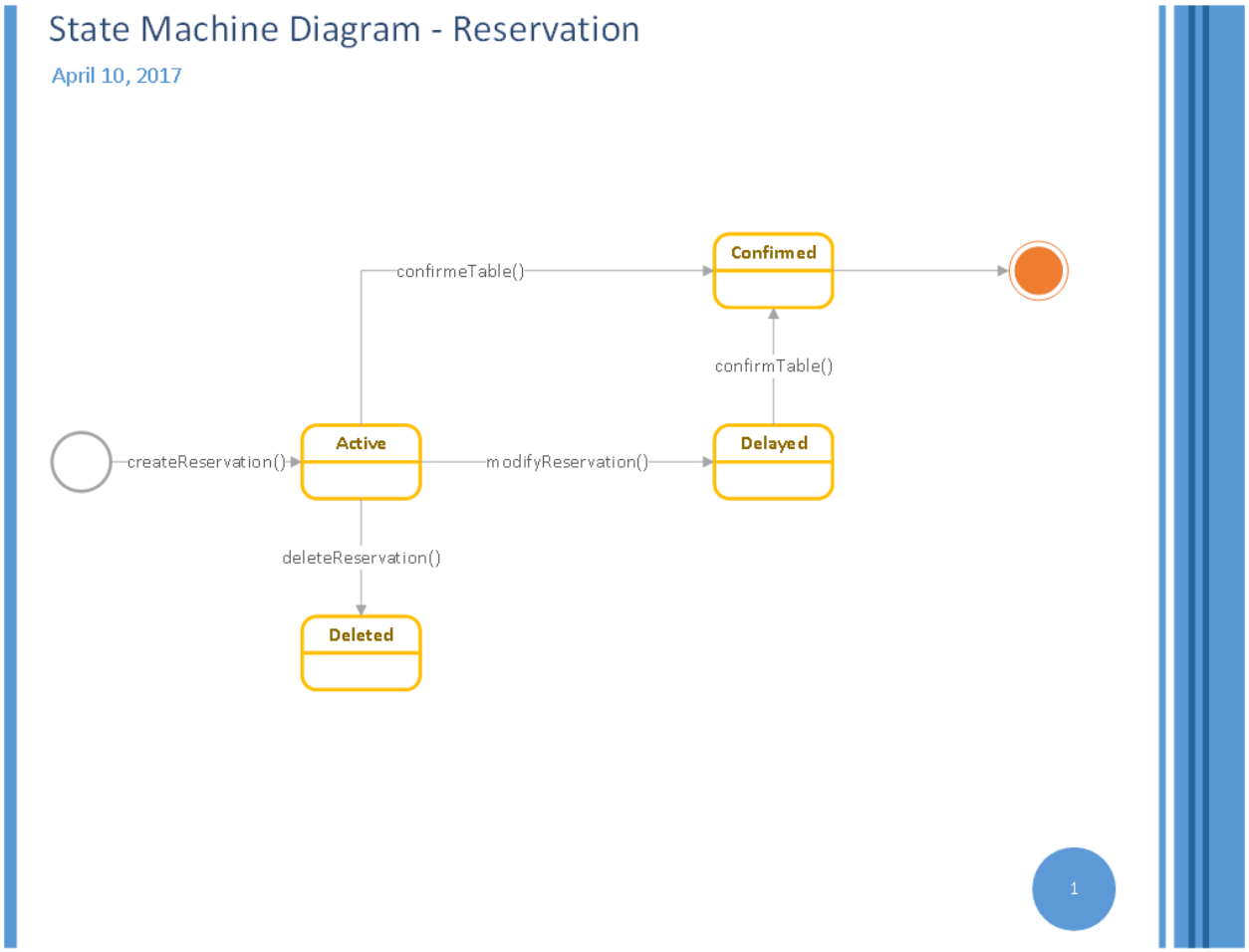


State Machine Diagram

Table

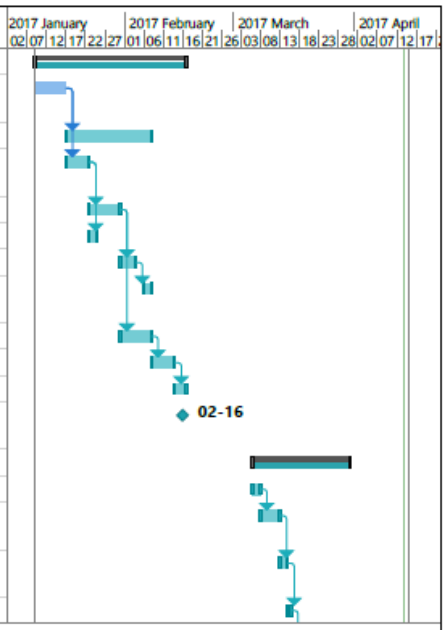


Reservation



Project Gantt Chart

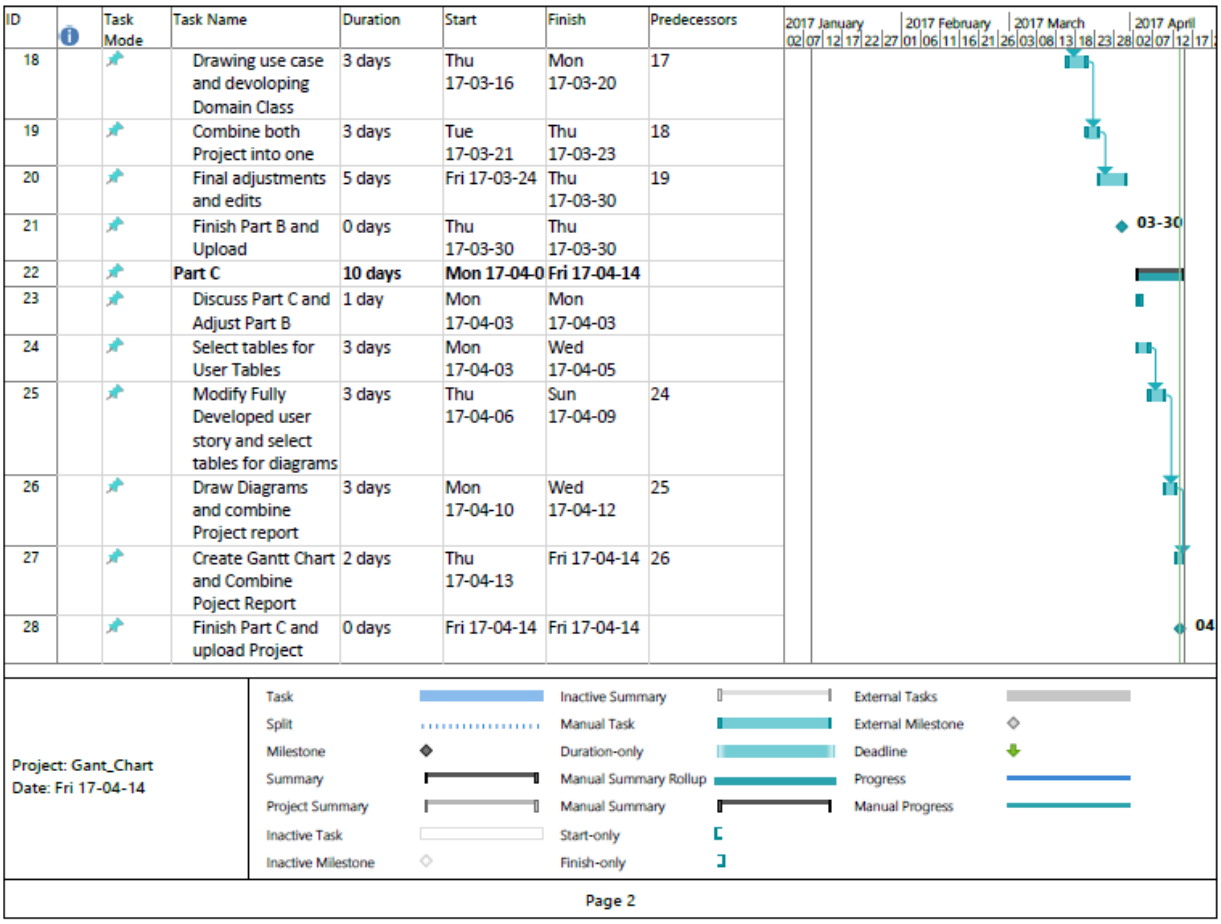
ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1		**Part A**	**29 days**	**Mon 17-01-0**	**Thu 17-02-16**	
2		Brainstorm and choosing an Idea	6 days?	Mon 17-01-09	Mon 17-01-16	
3		Creating Questiona	16 days	Tue 17-01-17	Tue 17-02-07	2
4		Identifying Stakeholders	5 days	Tue 17-01-17	Sun 17-01-22	2
5		Stakeholder Intervi	6 days	Mon 17-01-2	Mon 17-01-3	4
6		Draft 1	2 days	Mon 17-01-2	Tue 17-01-24	4
7		Draft 2	4 days	Tue 17-01-31	Fri 17-02-03	5
8		Making Diagrams and Tables	2 days	Mon 17-02-06	Tue 17-02-07	7
9		Draft 3 created	6 days	Tue 17-01-31	Tue 17-02-07	5
10		Draft 3 report	4 days	Wed 17-02-0	Mon 17-02-1	9
11		Final adjustments	3 days	Tue 17-02-14	Thu 17-02-16	10
12		Finish Part A and Upload	0 days	Thu 17-02-16	Thu 17-02-16	
13		**Part B**	**19 days**	**Mon 17-03-0**	**Thu 17-03-30**	
14		Dicussion of Part B	2 days	Mon 17-03-0	Tue 17-03-07	
15		Organzed Tables and wrote use	4 days	Wed 17-03-08	Sun 17-03-12	14
16		Organize tables and improve use	2 days	Mon 17-03-13	Tue 17-03-14	15
17		Combine Project	1 day	Wed 17-03-1	Wed 17-03-1	16



Project: Gant_Chart
Date: Fri 17-04-14

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1



ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors
18		Drawing use case and devoloping Domain Class	3 days	Thu 17-03-16	Mon 17-03-20	17
19		Combine both Project into one	3 days	Tue 17-03-21	Thu 17-03-23	18
20		Final adjustments and edits	5 days	Fri 17-03-24	Thu 17-03-30	19
21		Finish Part B and Upload	0 days	Thu 17-03-30	Thu 17-03-30	
22		**Part C**	**10 days**	**Mon 17-04-0**	**Fri 17-04-14**	
23		Discuss Part C and Adjust Part B	1 day	Mon 17-04-03	Mon 17-04-03	
24		Select tables for User Tables	3 days	Mon 17-04-03	Wed 17-04-05	
25		Modify Fully Developed user story and select tables for diagrams	3 days	Thu 17-04-06	Sun 17-04-09	24
26		Draw Diagrams and combine Project report	3 days	Mon 17-04-10	Wed 17-04-12	25
27		Create Gantt Chart and Combine Poject Report	2 days	Thu 17-04-13	Fri 17-04-14	26
28		Finish Part C and upload Project	0 days	Fri 17-04-14	Fri 17-04-14	

Appendix

Gantt Chart

Gant_Chart.pdf

Meeting Minutes

meeting_minutes.docx

meeting_minutes_PARTB.docx

meeting_minutes 3.docx

Stakeholder Register

STAKEHOLDER REGISTER					
STAKEHOLDER NAME	**Stakeholder Position**	**External/ Internal**	**Stakeholder contact details**	**Operational/ Executive**	**Interest (high, medium, low)**
BOB C,	Owner	External	Bobbafett99@gmail.com	Executive	Med
CHARLIE S.	Kitchen Manager	Internal	charleS@gmaill.com	Operational	Med
TINA G.	FoH Manager	Internal	tinag@gmail.com	Operational	High
LEE Y.	Line Cook	Internal	LeeY@gmail.com	Operational	Med
CHOI M.	Server	Internal	ChMk@gmail.com	Operational	Med

TALY H.	Host	Internal	talyyyy@gmail.com	Operational	High
CLARK K.	Guest	External	superman@gmail.com	Non-Operational	Low
LOU S.	Busser	Internal	LouS@gmail.com	Operational	Med

TableSmart Screenshots

Guest Seating:

Table Availability:

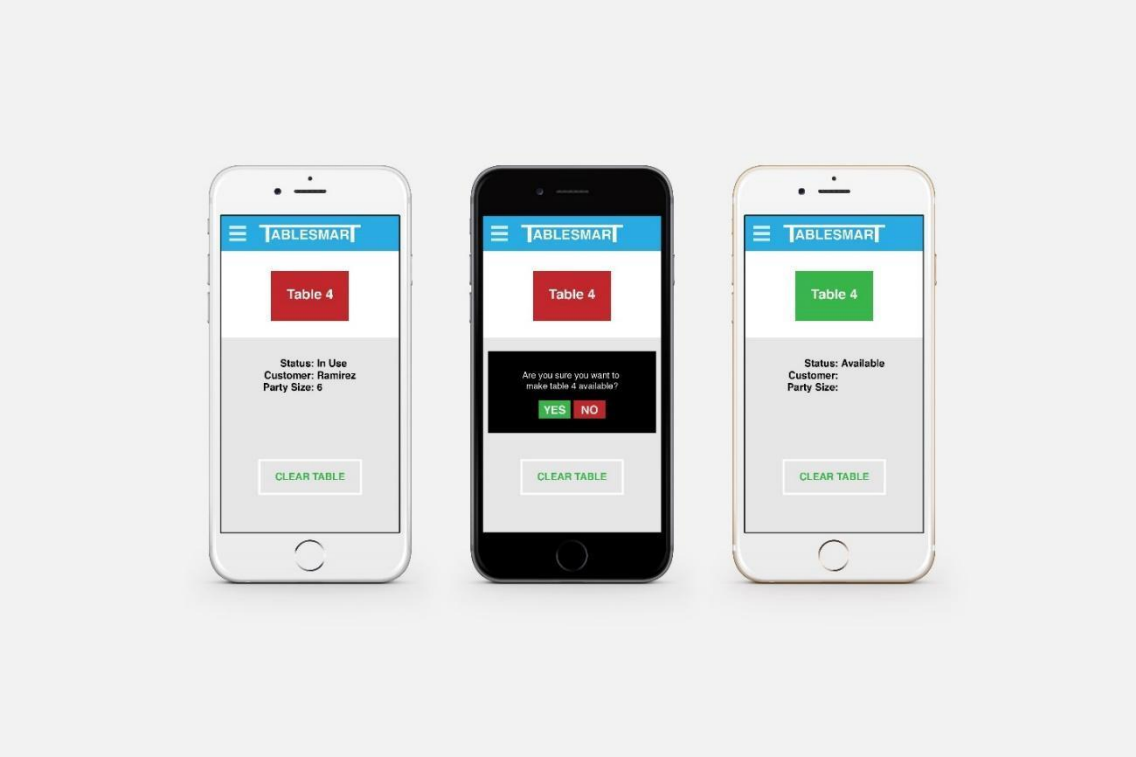


Table Queueing: