

ชื่อกลุ่ม :spades

ชื่อโปรเจค :ระบบบริหารจัดการโรงแรม

ชื่อของ GitHub repository: hotel-management-system

รายชื่อสมาชิก

1.นายณัฐกิจ ไพรสวรรณา 55010350

2.นายณัฐนนท์ รุ่งประเสริฐ 55010362

Object Oriented Analysis and Design, ภาคการศึกษาที่ 1/2557

บทคัดย่อ (Abstraction)

ในการบริหารจัดการโรงแรมนั้นมีความซับซ้อนวุ่นวายทั้งด้านการบริการลูกค้าโรงแรม , การเก็บประวัติลูกค้า , บันทึกการจองห้องพัก หรือ แม้กระทั่งการจัดการบุคลากรภายในตัวโรงแรม และ ในระบบจดบันทึกแบบดั้งเดิมนั้นไม่สามารถอำนวยความสะดวกในการทำงานได้มากพอและการค้นหาข้อมูลก็เป็นเรื่องที่ทำได้ยาก ระบบบริหารจัดการโรงแรมนี้จะเป็นตัวช่วยในการลดความยุ่งยากในการบริหารโรงแรมด้วยความสามารถในการแสดงรายละเอียดห้องพัก , เก็บประวัติลูกค้า , บันทึกการจองห้องพักและแสดงสถานะของต่างๆ ในรูปแบบปฏิทินไว้ในแอปพลิเคชันเดียว

งานที่เกี่ยวข้อง (Related works)

เป็นแอปพลิเคชันจะมีลักษณะใกล้เคียงกับ eZee Absolute , hotelopro

โดยมีข้อแตกต่าง คือ แอปพลิเคชันของเรารองรับพนักงานหลายตำแหน่งที่มีความต้องการเข้าถึงข้อมูลแตกต่างกัน และแอปพลิเคชันของเราไม่รองรับ online booking จากฝั่งลูกค้าเนื่องจากทางเราจะเน้นไปที่ การจัดการภายในเป็นหลัก

ผลการวิเคราะห์ความต้องการของผู้ใช้ระบบ

- ความต้องการที่เป็นหน้าที่หลัก (Functional requirement)

แสดงสถานะต่างๆของห้องได้ในรูปแบบปฏิทิน

สามารถทำการจองห้องพักได้

บันทึกรายละเอียดของผู้เข้าพัก

เพิ่ม/ลบ/แก้ไข รายละเอียดของห้อง

เพิ่ม/ลบ/แก้ไข พนักงานในโรงแรม

เพิ่ม/ลบ/แก้ไข รายละเอียดของโรงแรม

จัดการข้อมูลส่วนตัว

- ความต้องการที่ไม่ใช่หน้าที่หลัก (Non-functional requirement)

รองรับผู้ใช้หลายคนพร้อมกัน

ทำงานได้เร็ว – มีประสิทธิภาพสูง

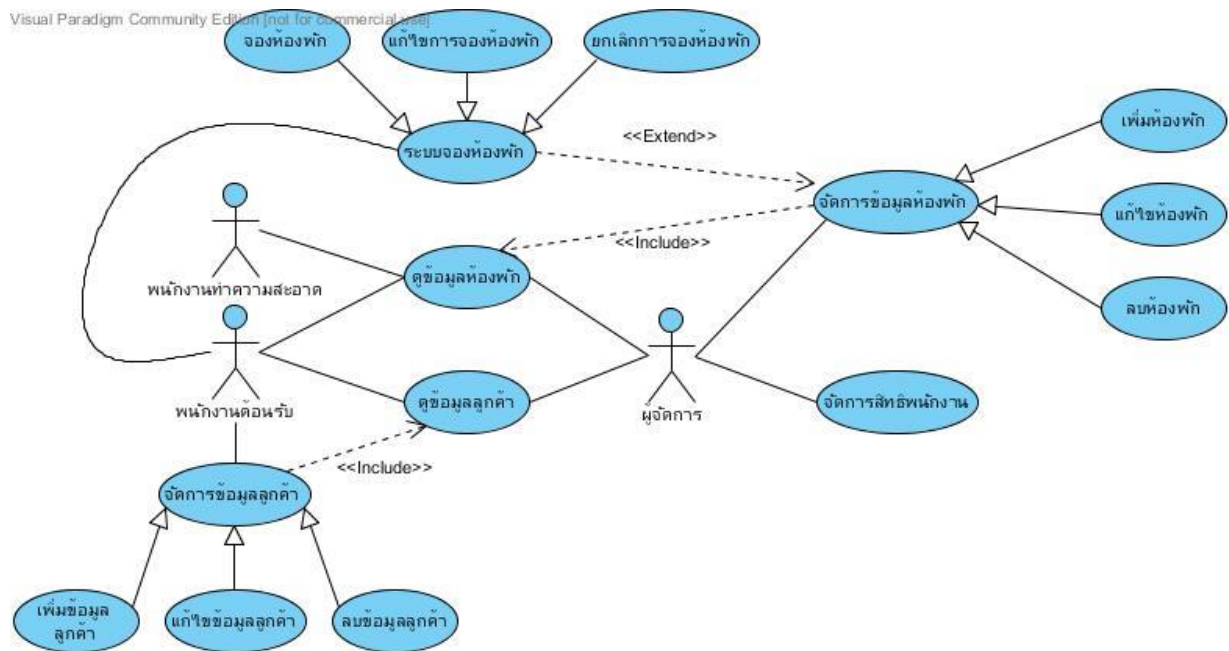
ใช้งานง่าย

พร้อมใช้งานตลอดเวลา

สามารถปรับปรุงแก้ไขระบบได้โดยง่าย

แนวทางการใช้งาน

Visual Paradigm Community Edition [not for commercial use]



รายละเอียดกรณีการใช้งาน (Use Case Specification)

ชื่อกรณี (Use case name)

การจองห้องพัก (reservation)

วัตถุประสงค์ (Use case purpose)

จองห้องโดยใช้พนักงานต้อนรับเป็นผู้ทำการเลือกห้องพักให้กับลูกค้าที่เข้าพักโดยการจองสามารถถูกยกเลิกแก้ไขได้เช่นกัน

ตัวแปรต้น (Preconditions)

ต้องเข้าระบบและอยู่ในตำแหน่งพนักงานต้อนรับ

ตัวแปรตาม (Postconditions)

จองห้องพักเสร็จสิ้น

ข้อจำกัด (Limitations)

ไม่สามารถจองห้องพักออนไลน์ (online booking) จากฝั่งลูกค้าได้

สมมติฐาน (Assumptions)

การจองห้องพักต้องรู้เลขห้องและจำนวนวันที่จอง

สถานการณ์หลัก (Primary scenario)

1. ลูกค้าตัดสินใจเกี่ยวกับประเภทห้องพักและจำนวนวันที่เข้าพัก
2. พนักงานเลือกหมายเลขห้องที่ตรงกับความต้องการของลูกค้า
3. ใส่จำนวนวันที่ลูกค้าต้องการ
4. ยืนยันการจอง
5. จบการทำงาน

สถานการณ์อื่น (Alternate scenario)

เงื่อนไขที่ทำให้เกิดสถานการณ์อื่น (Condition triggering alternate flow scenario)

เงื่อนไขที่ 1 ห้องพักที่ลูกค้าต้องการถูกจองเต็มทุกห้องในวันที่ลูกค้าต้องการ

- 1-1. พนักงานแจ้งลูกค้าว่าห้องที่ต้องการเต็ม
- 1-2. เสนอห้องที่มีลักษณะและราคาใกล้เคียงกับความต้องการของลูกค้าและไม่ถูกจองในวันที่ลูกค้าต้องการ
 - 2-1. ถ้าลูกค้าตกลงดำเนินสถานการณ์หลัก (Primary scenario) ต่อในขั้นตอนที่ 2.
 - 2-2. ถ้าลูกค้าไม่ตกลงแต่ต้องการจองห้องพัก ให้ไปขั้นตอนที่ 1-2.
 - 2-3. ถ้าลูกค้าไม่ตกลงและไม่ต้องการจองห้องพัก ให้ถือว่าไม่มีการจองเกิดขึ้น
 - 2-4. จบการทำงาน

ชื่อกรณี (Use case name)

เพิ่มห้องพัก

วัตถุประสงค์ (Use case purpose)

ให้ผู้จัดการสามารถเพิ่มห้องพักของและใส่รายละเอียดต่างๆได้

ตัวแปรต้น (Preconditions)

เข้าระบบในฐานะผู้จัดการของโรงแรม

ตัวแปรตาม (Postconditions)

เพิ่มห้องพักพร้อมทั้งรายละเอียดของห้อง

ข้อจำกัด (Limitations)

ไม่สามารถเพิ่มห้องที่หลายๆห้องที่มีรายละเอียดต่างกันพร้อมกันได้

สมมติฐาน (Assumptions)

ผู้จัดการทราบรายละเอียดทั้งหมดที่จำเป็นของห้องที่ต้องการเพิ่ม

สถานการณ์หลัก (Primary scenario)

1. ผู้ใช้เข้าสู่ระบบในฐานะผู้จัดการโรงแรม
2. ผู้จัดการโรงแรมทำการเพิ่มห้องพัก
3. ใส่ประเภทของห้อง, ราคา, จำนวนเตียง, หมายเลขห้อง, ชั้นและรายละเอียดอื่นๆ
4. ยืนยันการเพิ่มห้อง
5. จบการทำงาน

สถานการณ์อื่น (Alternate scenario)

เงื่อนไขที่ทำให้เกิดสถานการณ์อื่น (Condition triggering alternate flow scenario)

เงื่อนไขที่ 1: กรอกข้อมูลไม่ครบในขั้นตอนการเพิ่มห้องพัก

3-1. ผู้จัดการกรอกรายละเอียดในส่วนต่างๆไม่ครบ

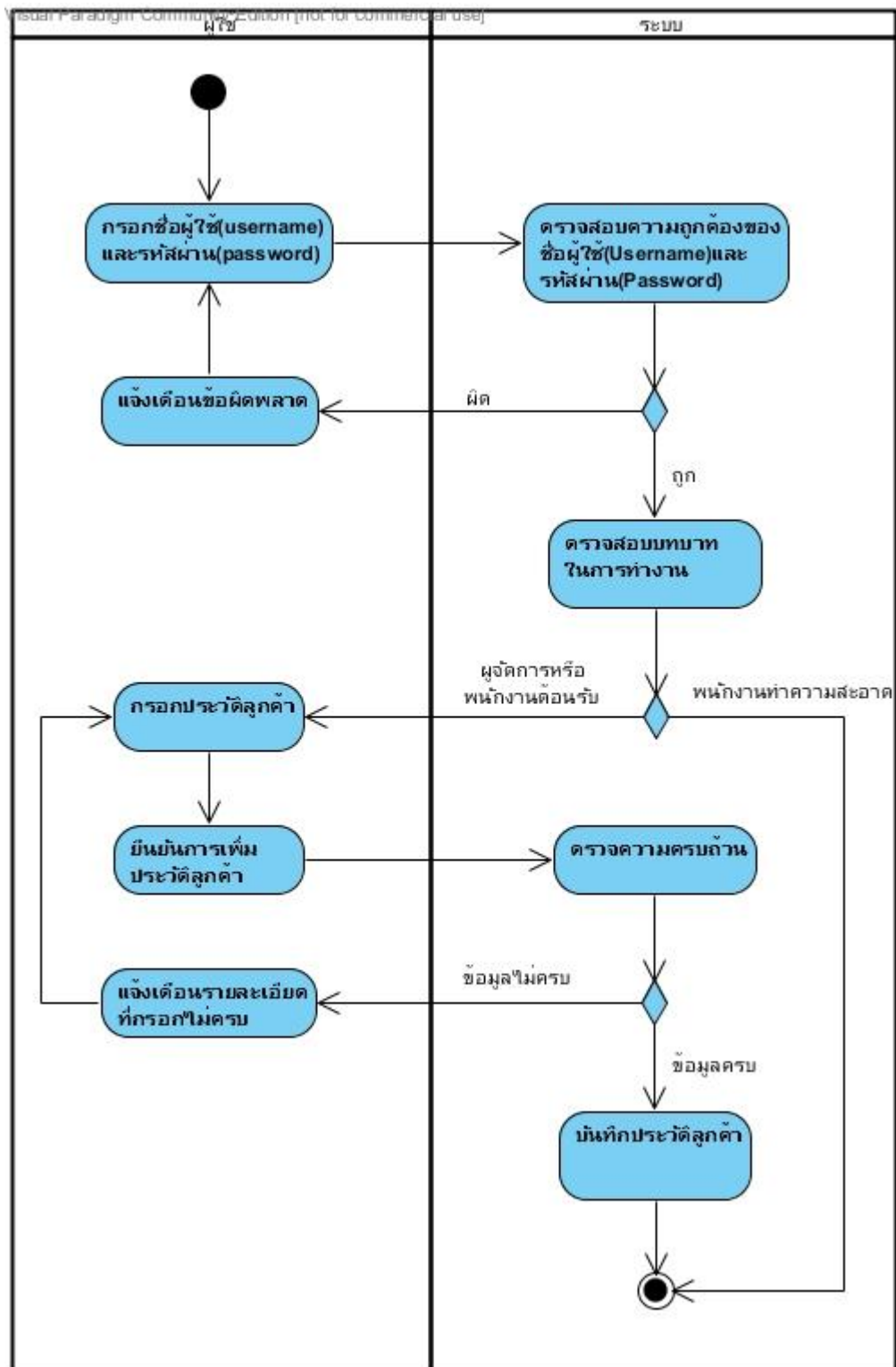
3-2. ระบบแจ้งเตือนรายละเอียดที่กรอกไม่ครบ

3-3. ผู้จัดการกรอกรายละเอียดส่วนที่ขาดหาย

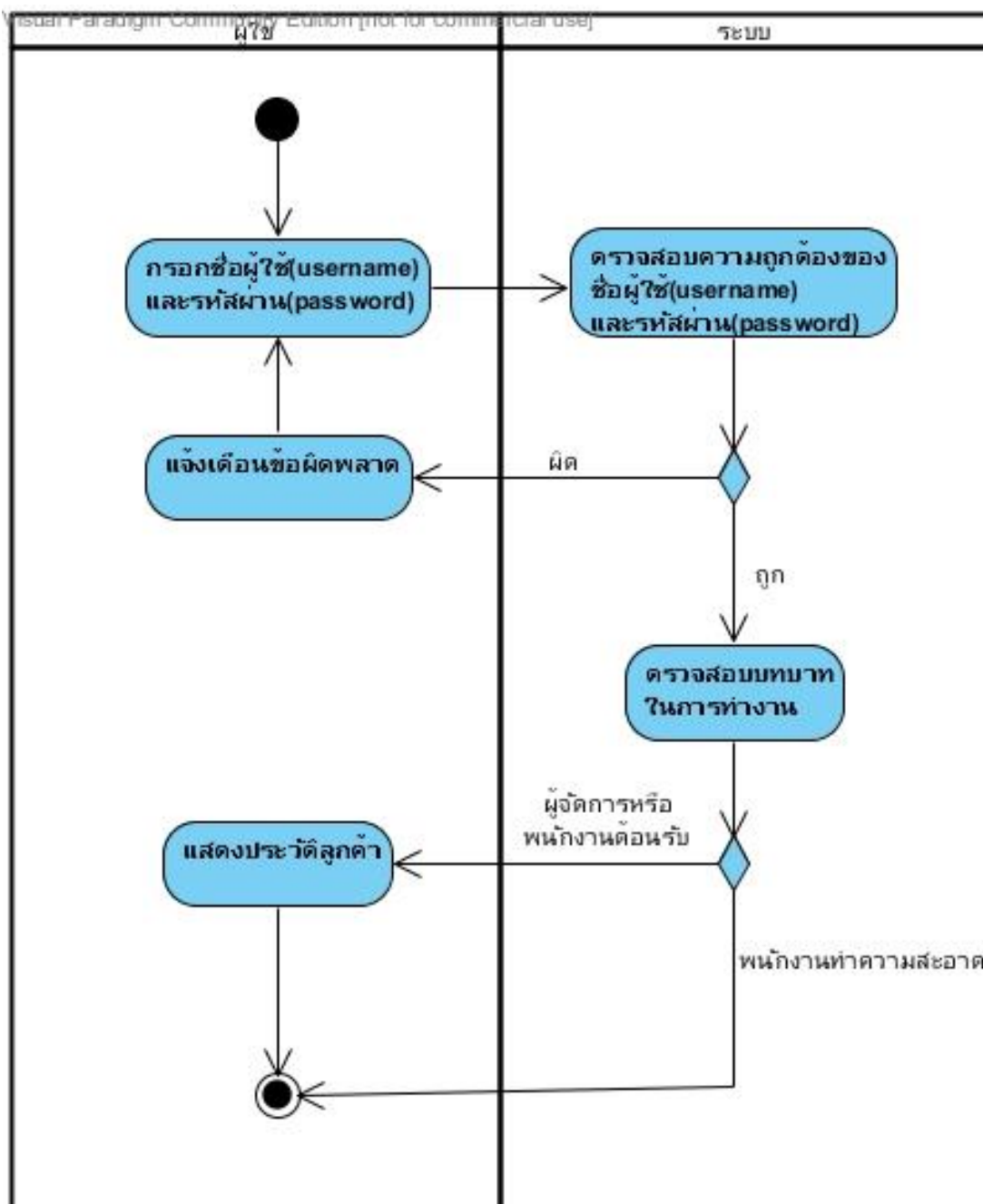
3-4. ดำเนินสถานการณ์หลัก (Primary scenario) ต่อในขั้นตอนที่ 4.

Activity Diagram

เพิ่มประวัติลูกค้า



ดูข้อมูลลูกค้า



สถาปัตยกรรมของระบบ

องค์ประกอบ(Component)

ระบบจัดการห้อง

เพิ่มห้อง เพิ่มรายชื่อห้องและรายละเอียดต่างๆของห้อง

แก้ไขรายละเอียดห้องที่สร้างไว้แล้ว

ลบห้องจากฐานข้อมูล

ระบบจัดการสถานะห้อง

เปลี่ยนสถานะห้อง

ห้องว่าง

จอง บันทึกหมายเลขห้อง และ จำนวนวันที่ต้องการจอง

มีคนเข้าพัก บันทึกหมายเลขห้อง และ จำนวนวันที่ต้องการจอง

ปิดปรับปรุง บันทึกหมายเลขห้อง และ จำนวนวันที่ต้องการจอง

ระบบจัดการแขก

เพิ่มแขก เพิ่มรายละเอียดของแขกที่เข้าพักประกอบด้วย ชื่อ, เพศ, ที่อยู่, รหัสบัตรประจำตัวประชาชนหรือเลขหนังสือเดินทาง, วันเดือนปีเกิด และสัญชาติ

แก้ไขรายละเอียดแขกที่สร้างไว้แล้ว

ลบแขกออกจากฐานข้อมูล

ระบบจัดการพนักงาน

เพิ่มพนักงาน เพิ่มชื่อ-นามสกุล และที่อยู่ติดต่อของพนักงาน รวมทั้งสิทธิ์ในการเข้าถึงข้อมูลต่างๆ

จัดการสิทธิ์ของพนักงานในการดูและแก้ไขข้อมูลต่างๆ เช่น รายละเอียดแขก, สถานะห้อง
ลบพนักงานออกจากระบบ

การใช้งาน(Deployment)

VM Window 2012 R2 Data Center Server IP address 23.101.22.3

Laravel php Framework ที่ใช้พัฒนา

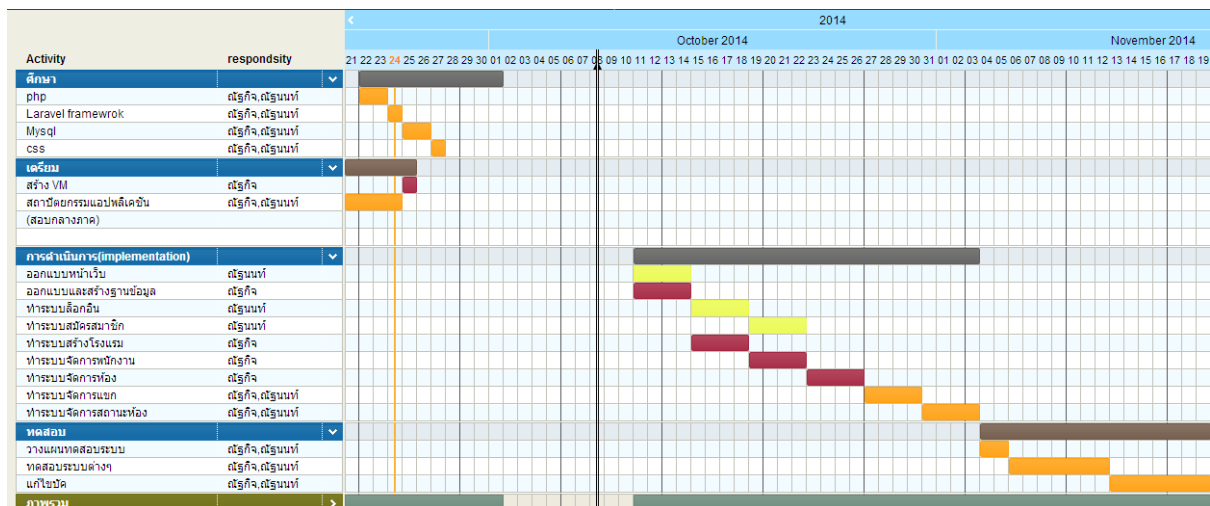
Bootstrap เป็น Frontend Framework

MySQL เป็นโปรแกรมที่ใช้ในการจัดการเกี่ยวกับฐานข้อมูล

XAMPP Apache web server เพื่อจำลองการทำงาน

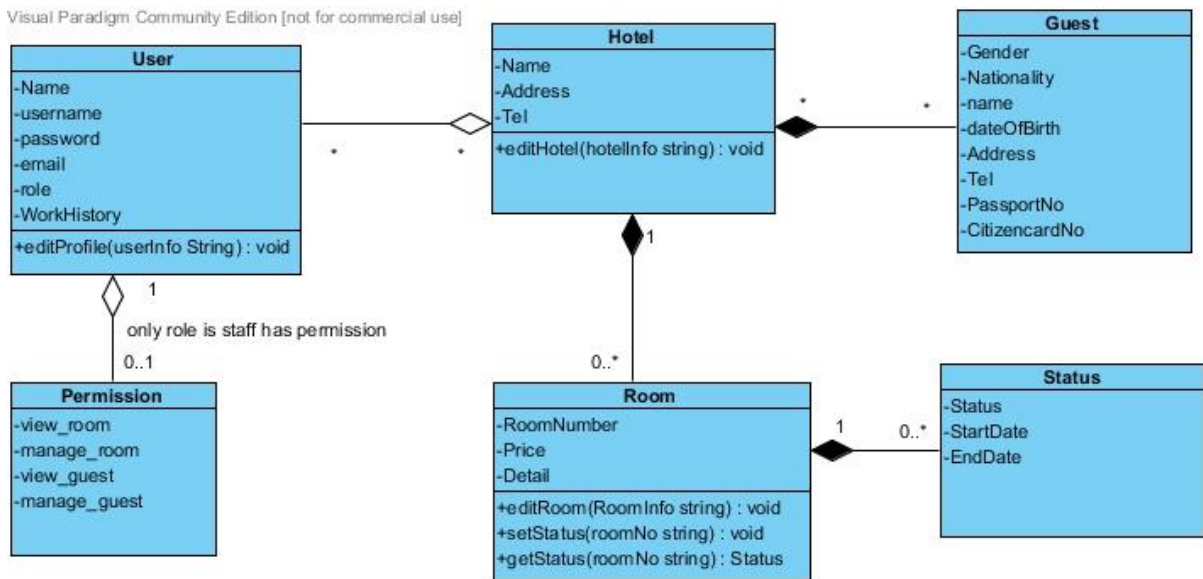
phpMyAdmin เป็นตัวช่วยจัดการฐานข้อมูล

รายละเอียดการพัฒนาซอฟต์แวร์ (Implementation Plan)



แผนภาพของคลาสหลัก (Class diagram)

Visual Paradigm Community Edition [not for commercial use]



ผลการทดสอบซอฟต์แวร์

Unit Test

Directory : /app/tests/HotelTest.php

ทดสอบการสร้างโรงแรม

ทดสอบการสร้างโรงแรมด้วยข้อมูลที่ซ้ำ

Directory : /app/tests/RoomTest.php

ทดสอบการสร้างสถานะห้อง

Directory : /app/tests/PermissionTest.php

ทดสอบระบบจำกัดสิทธิ์พนักงาน

Evaluation

ระบบจำกัดสิทธิ์พนักงาน

จุดประสงค์การทดลอง

เพื่อตรวจสอบว่าผู้จัดการสามารถจำกัดสิทธิ์พนักงานได้จริง

สิ่งที่จะวัด

การแสดงผลของหน้าเว็บที่ล็อกอินด้วย id ของพนักงานที่ถูกจำกัดสิทธิ์ไว้

วิธีการทดลอง

ผู้จัดการ สร้างโรงแรมของตัวเองและ สร้าง User 1 คนสำหรับสมัครเป็นพนักงานของโรงแรม

User ส่งคำขอเป็น staff ของโรงแรมดังกล่าว

Manager ตอบรับคำขอเป็น staff ของ userคนนั้น

Manager กำหนดสิทธิ์ staff แตกต่างกันในแต่ละรอบ

Table permission Test

จำนวนรอบ	Room permission			Guest Permission		
	No room	View room	Manage room	No guest	View guest	Manage guest
1	1	0	0	1	0	0
2	0	1	0	0	1	0
3	0	0	1	0	0	1

สิ่งที่ต้องใช้ในการทดลอง

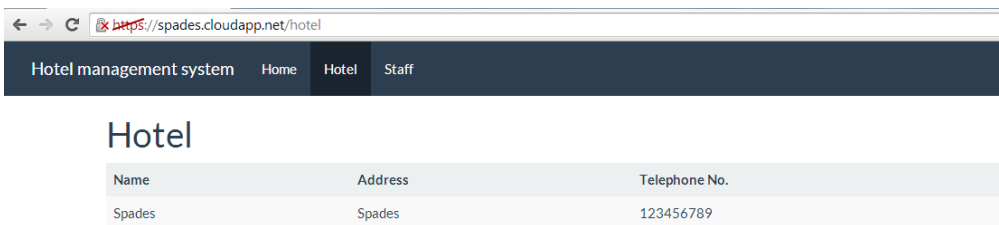
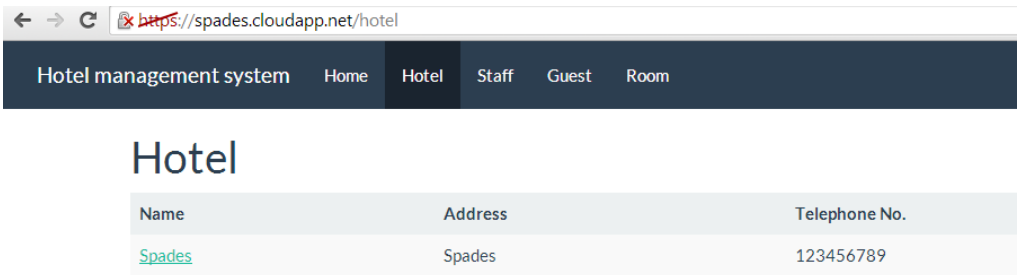
ข้อมูลของโรงแรมในฐานะข้อมูล

ข้อมูลของผู้ใช้งานในฐานะข้อมูล

ข้อมูลหน้าที่ของผู้ใช้งานในฐานะข้อมูล (manager,staff)

ผลที่ได้จากการทดลอง

1. รอบแรก พนักงานไม่สามารถดูข้อมูลของ room และ guest ได้
2. รอบสอง พนักงานสามารถดูข้อมูล ของ room และ guest ได้
3. รอบสาม พนักงานสามารถจัดการข้อมูล ของ room และ guest ได้

จำนวน รอบ	Staff view
1	<p>รอบ 1 : staff ไม่ได้รับอนุญาต ให้เข้าถึง room และ guest ไม่มีแถบรายชื่อให้เลือก</p> 
2	<p>รอบ 2 : staff ได้รับอนุญาต ให้ดู room และ guest ได้จึงมีแถบรายชื่อเพิ่มขึ้นมา และ มี link ของ hotel เพื่อเข้าดูปฏิทินของสถานะห้อง</p> 

← → ↻ <https://spades.cloudapp.net/hotel/5>

Hotel management system Home Hotel Staff Guest Room

Occupied Reserved Maintenance

< > today November 2014 month week day

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21 Spades	22 Spades
23 Spades	24	25	26	27	28	29

หน้าปฏิทินของสถานะห้อง

กรณีนี้ผู้ใช้จะไม่สามารถลบสถานะ ห้องทาง ปฏิทินได้ และเมื่อทำการลบ จะทำการแสดงข้อความแจ้งเตือนและนำผู้ใช้ กลับไปหน้า home

← → ↻ <https://spades.cloudapp.net/hotel/5>

Hotel management system Home Hotel Staff Guest

หน้าเว็บที่ https://spades.cloudapp.net แจ้งว่า

Are you sure you want to delete this event?

ลบออก ยกเลิก

Sun	Mon	Tue	Wed	Thu	Fri	Sat
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21 Spades	22 Spades
23 Spades	24	25	26	27	28	29
30	1	2	3	4	5	6

← → ↻ <https://spades.cloudapp.net>

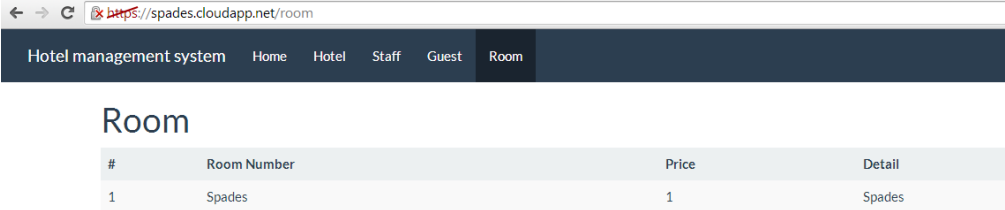
Hotel management system Home Hotel Staff Guest Room

Error

Access deny

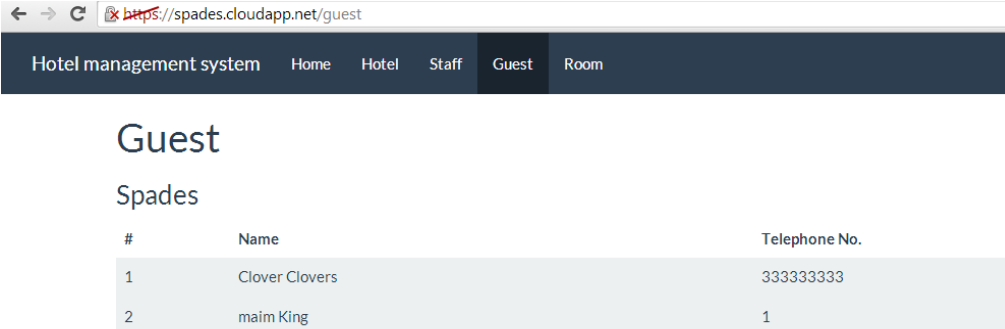
Hotel management system

เมื่อคลิกแถบ Room แสดงรายละเอียดต่างๆของห้อง



#	Room Number	Price	Detail
1	Spades	1	Spades

เมื่อคลิกแถบ Guest แสดงรายชื่อของแขก

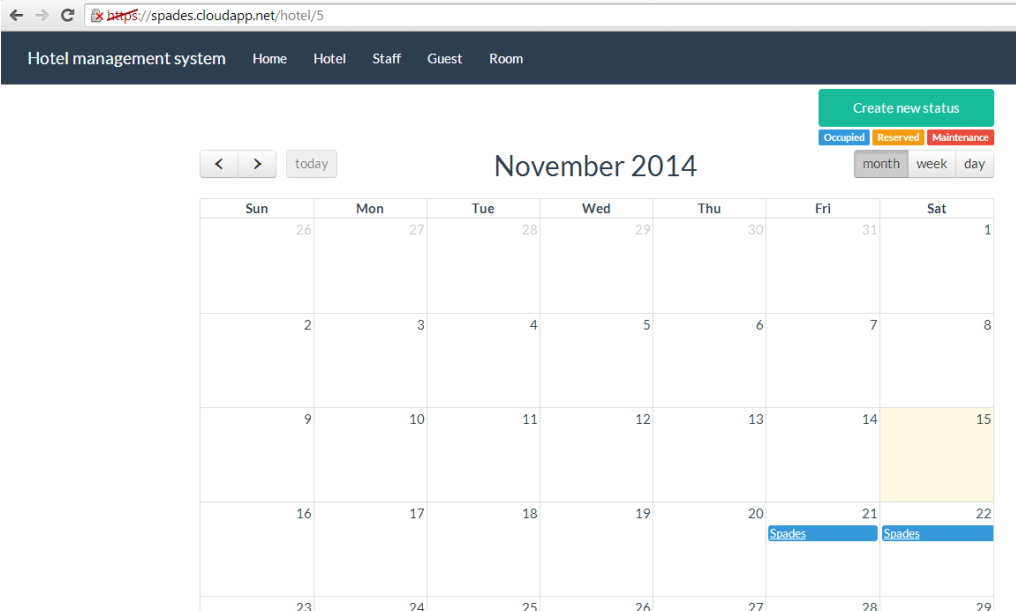


#	Name	Telephone No.
1	Clover Clovers	333333333
2	maim King	1

3

รอบ 3 : Staff ได้รับอนุญาต ให้ manage room และ manage guest

Manage room : สามารถดู และ ทำการเพิ่ม/ลบสถานะห้องได้



Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

หน้าปฏิทิน เพิ่ม ปุ่ม Create new status จากรอบที่2 ซึ่งใช้ในการสร้างสถานะห้อง

นอกจากนี้สามารถลบ Status ทางตารางปฏิทินได้

← → ↻ <https://spades.cloudapp.net/hotel/5>

Hotel management systemHomeHotelStaffGuestRoom

หน้าเว็บที่ https://spades.cloudapp.net แจ้งว่า:
Are you sure you want to delete this event?

ลบออกยกเลิก

	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	29
	30	1	2	3	4	5	6

← → ↻ <https://spades.cloudapp.net/hotel/5>

Hotel management systemHomeHotelStaffGuestRoom

Create new status

OccupiedReservedMaintenance

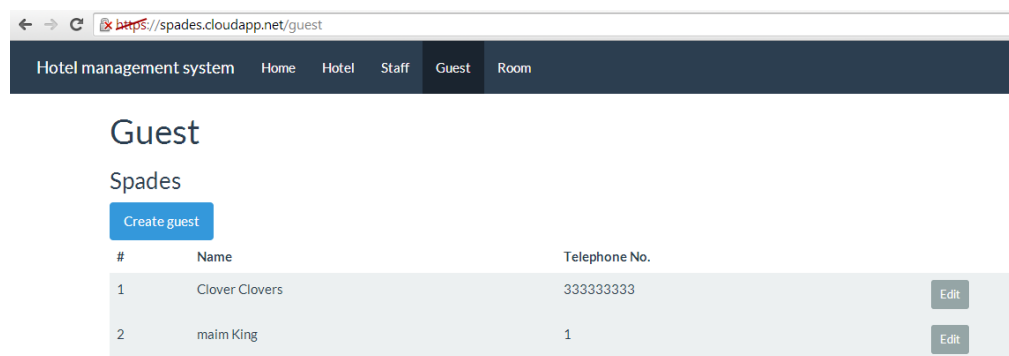
< > today

November 2014

monthweekday

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

Manage guest : สามารถดู และ ทำการเพิ่ม/แก้ไข แยกได้



สรุปและสิ่งที่ได้จากการทดลองนี้

ผู้จัดการสามารถจำกัดสิทธิ์พนักงานได้จริง

ระบบจัดการสถานะห้องพัก

จุดประสงค์การทดลอง

ห้องพักสามารถแสดงสถานะของห้องตามช่วงเวลาได้ถูกต้อง และ ระบบการสร้างสถานะของห้องสามารถทำงานได้ถูกต้อง

สิ่งที่วัด

การแสดงผลของสถานะต่างๆจากปฏิทิน

วิธีการทดลอง

มี manager 1 คนสร้างโรงแรม 1 โรงแรม และเพิ่มห้อง 1 ห้องโดยการสร้างสถานะห้องต้องเข้าเงื่อนไขดังนี้

1. ไม่สามารถสร้างสถานะย้อนหลังวันปัจจุบันได้
2. ไม่สามารถตั้งวันสิ้นสุด (End date) ก่อนวันเริ่ม วันเริ่มต้นได้ (Start date)
3. ไม่สามารถตั้งวันเริ่มต้น (Start date) เป็นวันเดียวกับ วันสิ้นสุดได้ (End date)
4. ไม่สามารถสร้างสถานะได้ ถ้าช่วงเวลาดังกล่าวมีสถานะอยู่แล้ว
5. เมื่อสร้างสถานะห้องได้แล้วเราสามารถลบสถานะห้องได้เช่นกัน

ดังนั้นจึงออกแบบการทดลองดังนี้

1. ทำการสร้างสถานะเป็น Maintenance วันที่ 14/11/2014 ถึง วันที่ 22/11/2014 และตรวจสอบผลลัพธ์
2. ทำการสร้างสถานะเป็น Occupied วันที่ 22/11/2014 ถึง วันที่ 21/11/2014 และตรวจสอบผลลัพธ์
3. ทำการสร้างสถานะเป็น Occupied วันที่ 21/11/2014 ถึง วันที่ 21/11/2014 และตรวจสอบผลลัพธ์

- 4.1 ทำการสร้างสถานะเป็น Occupied วันที่ 21/11/2014 ถึง วันที่ 22/11/2014 และตรวจสอบผลลัพธ์
- 4.2 ทำการสร้างสถานะเป็น Reserved วันที่ 22/11/2014 ถึง วันที่ 29/11/2014 และตรวจสอบผลลัพธ์
- 4.3 ทำการสร้างสถานะเป็น Maintenance วันที่ 21/11/2014 ถึง วันที่ 30/11/2014 และตรวจสอบผลลัพธ์
5. ทำลบสถานะ Reserved ในข้อ 4.2 และ สร้างสถานะ Occupied ในช่วงเวลาเดียว กับข้อ 4.2 และตรวจสอบผลลัพธ์

สิ่งที่ต้องใช้ในการทดลอง

ข้อมูลของโรงแรมในฐานะข้อมูล

ข้อมูลของห้องพักในฐานะข้อมูล

ข้อมูลสถานะห้องในฐานะข้อมูล

ข้อมูลของผู้ใช้งานในฐานะข้อมูล

ผลที่ได้จากการทดลอง

1.ไม่สามารถสร้างได้พร้อมแสดงข้อความแจ้งเตือน

Hotel management system Home Hotel Room Guest Staff Request Logout Spades

Create new room event

Room:

Status:

Start date:
The start date must be a date after 2014-11-14.

End date:

2.ไม่สามารถสร้างได้พร้อมแสดงข้อความแจ้งเตือน

Hotel management system Home Hotel Room Guest Staff Request Logout Spades

Create new room event

Room:

Status:

Start date:

End date:
The end date must be a date after start date.

3.ไม่สามารถสร้างได้พร้อมแสดงข้อความแจ้งเตือน

Hotel management system Home Hotel Room Guest Staff Request Logout Spades

Create new room event

Room: Spades

Status: Occupied

Start date: 2014-11-21

End date: 2014-11-21

The end date must be a date after start date.

Submit cancel

4.1สามารถสร้างได้ พร้อมแสดงข้อความ success

Hotel management system Home Hotel Room Guest Staff Request Logout Spades

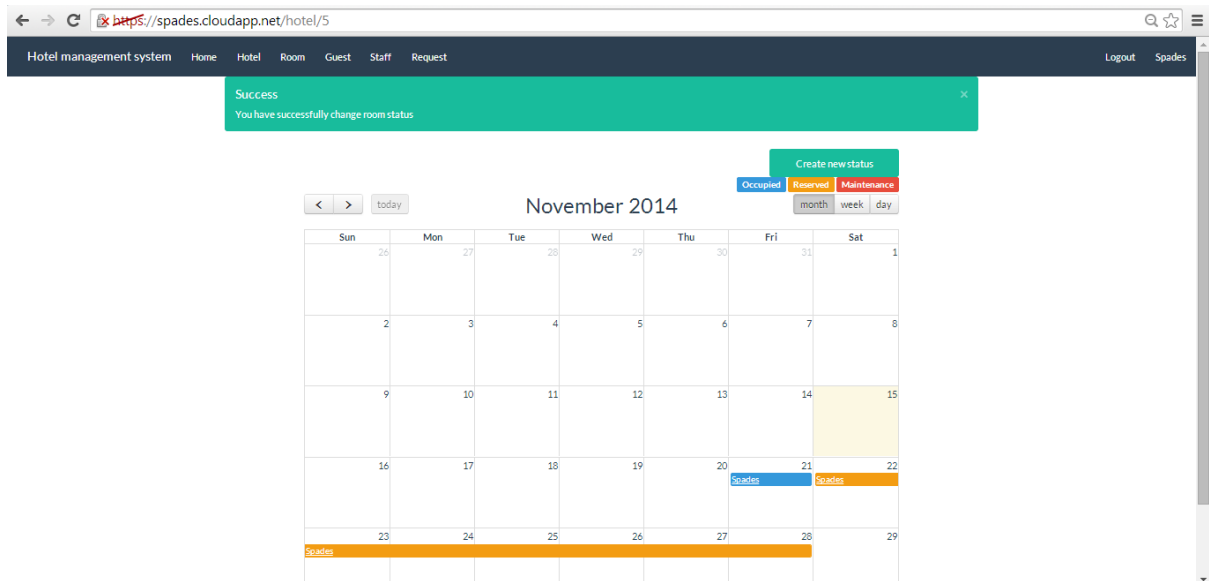
Success
You have successfully change room status

Create new status
Occupied Reserved Maintenance
month week day

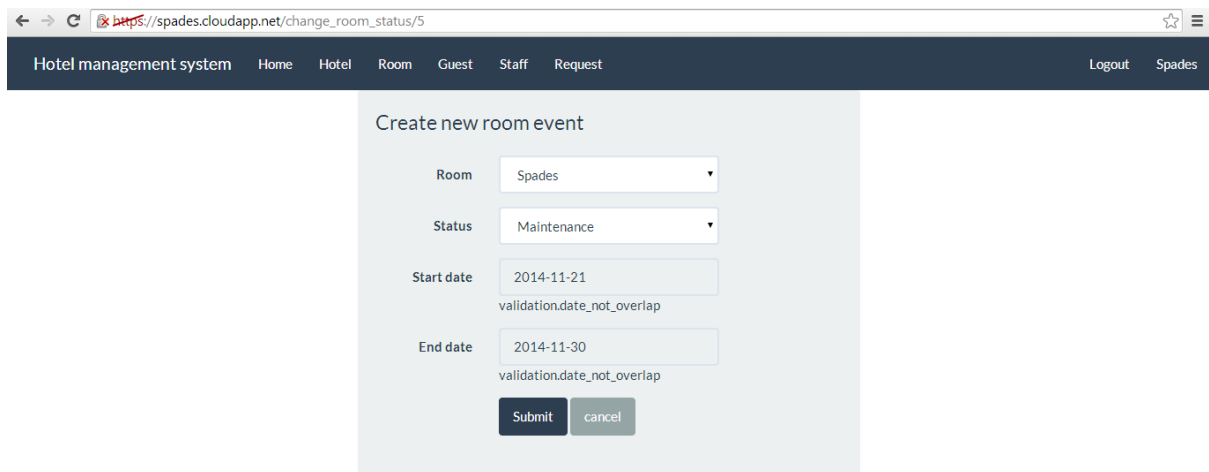
November 2014

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21 Spades	22
23	24	25	26	27	28	29

4.2สามารถสร้างได้ พร้อมแสดงข้อความ success

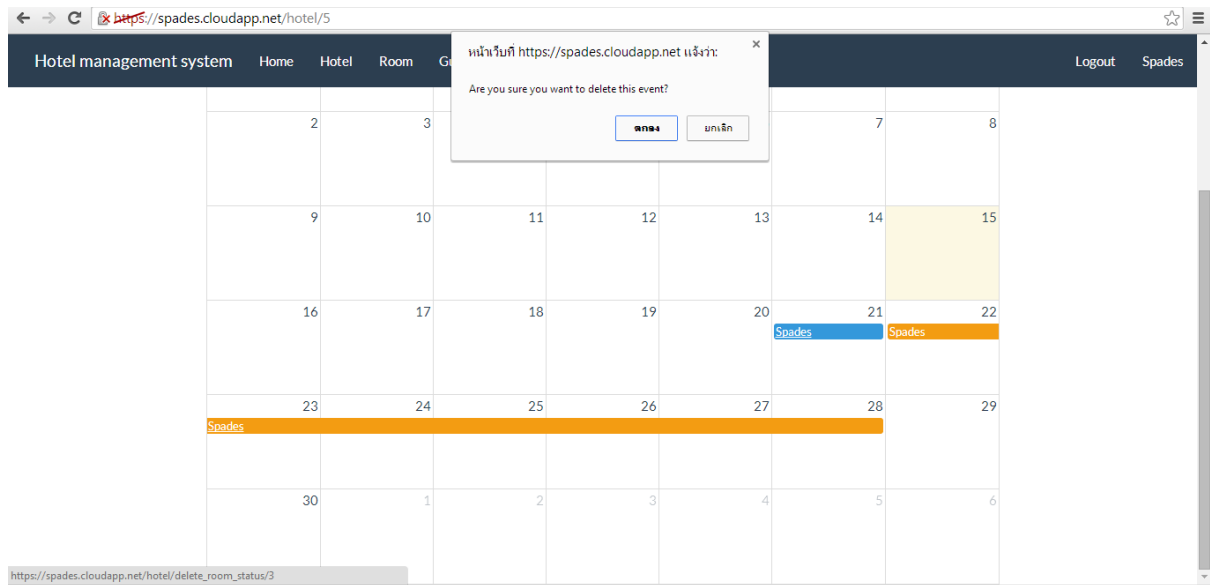


4.3ไม่สามารถสร้างได้ พร้อมแสดงข้อความแจ้งเตือน

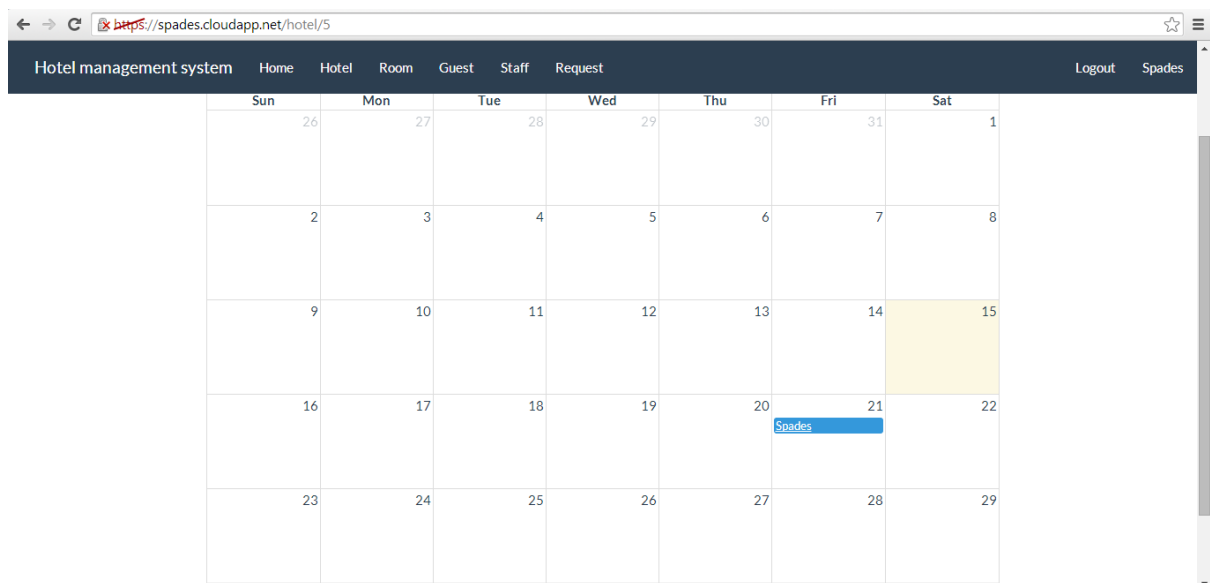


5. สามารถทำได้ พร้อมแสดงขั้นตอนดังนี้

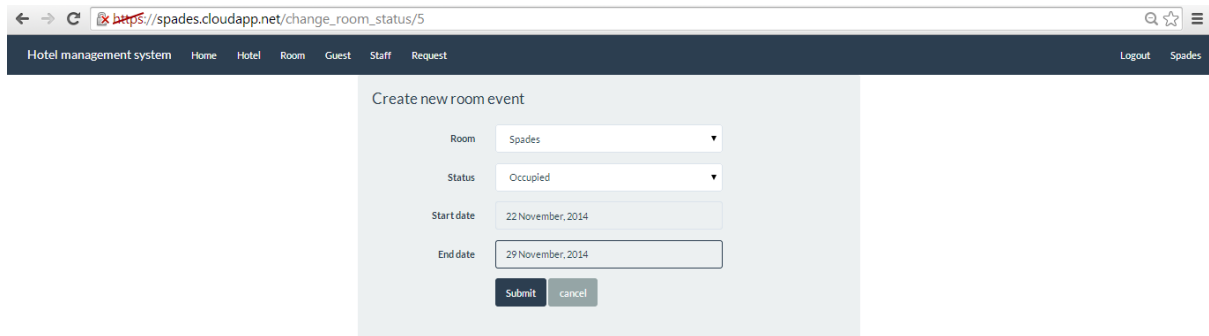
5.1 แสดงข้อความยืนยันการลบ สถาน



ผลของข้อ5.1



5.2 ทำการสร้างสถานะใหม่ในช่วงเวลาเดียวกับสถานะที่เพิ่งลบไป



Hotel management system Home Hotel Room Guest Staff Request Logout Spades

Create new room event

Room Spades

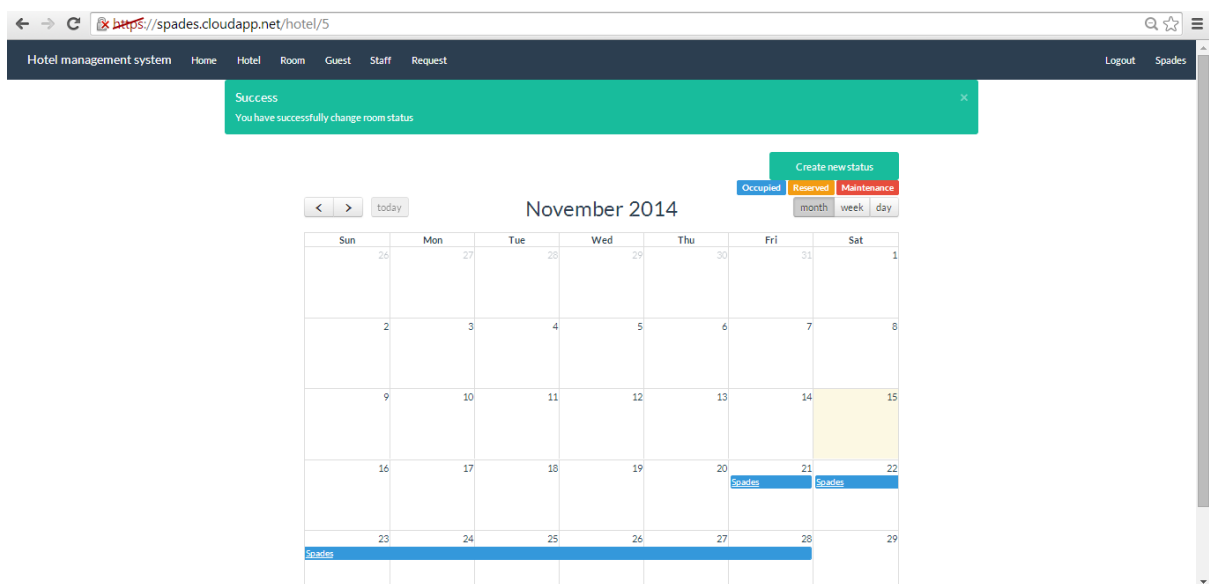
Status Occupied

Start date 22 November, 2014

End date 29 November, 2014

Submit cancel

ผลลัพธ์ของข้อ5.2



สรุปและสิ่งที่ได้จากการทดลองนี้

ห้องพักสามารถแสดงสถานะของห้องพักตามช่วงเวลาได้ถูกต้อง และระบบการสร้างสถานะของห้องพักสามารถทำงานได้ถูกต้องตามข้อกำหนดที่ได้ตั้งไว้

บทสรุป

ระบบบริหารจัดการโรงแรมสามารถทำงานได้ตาม functional requirement ที่ระบุไว้และมีความสามารถตามความต้องการขั้นต่ำของระบบที่ได้ระบุไว้

บรรณานุกรม(Bibliography)

@mdo and @fat (2011-2014) Bootstrap

<http://getbootstrap.com/>

Thomas Park (2013) Bootstrap's flatly theme

<http://bootswatch.com/flatly/>

Adam Shaw (2014) jQuery plugin FullCalendar

<http://fullcalendar.io/>

Amsul (2013) jQuery plugin pickadate

<http://amsul.ca/pickadate.js/>