



รายงาน

Project phase 2 : Fitness Center System

โดย

6687015	นางสาวฐิตารีย์	กองแก้ว
6687019	นางสาวดารากร	แซ่ภู
6687031	นางสาวพรปวีณ์	ปฐมพรวิวัฒน์
6687057	นางสาวสุธารทิพย์	หลวงทิพย์
6687093	นางสาวภัทรกร	ต้นสมบูรณ์

เสนอ

ผศ.ดร.ฉันทิ์ สุเนตนันท์

ผศ.ดร. ชัยยงค์ รักจิตเวชสกุล

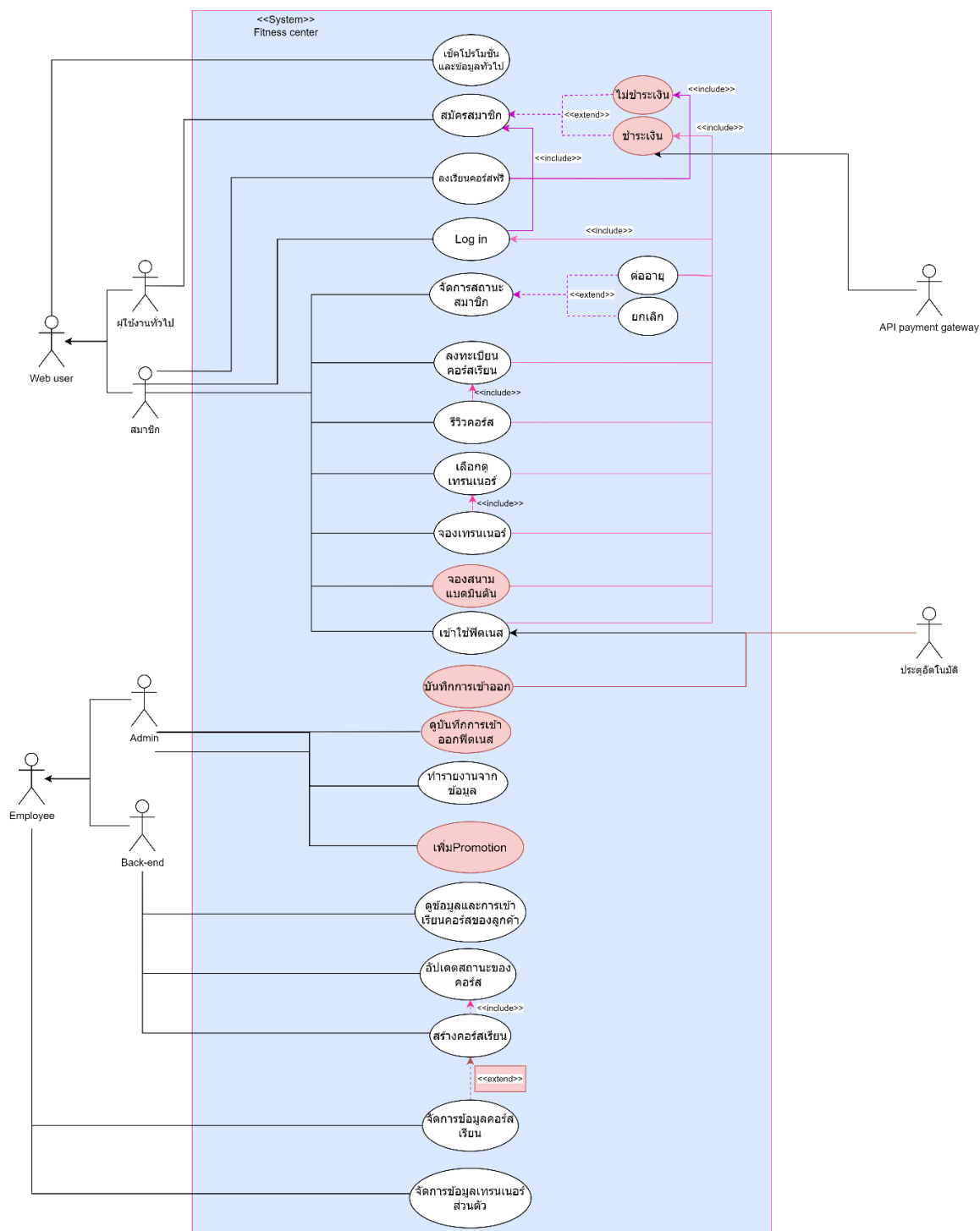
น.ส. ภัทรพร แสงอรุณศิลป์

ภาคเรียนที่ 1 ปีการศึกษา 2567

คณะเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยมหิดล

รายการแก้ไขของ Project Phase 1

1.แก้ไข Use case

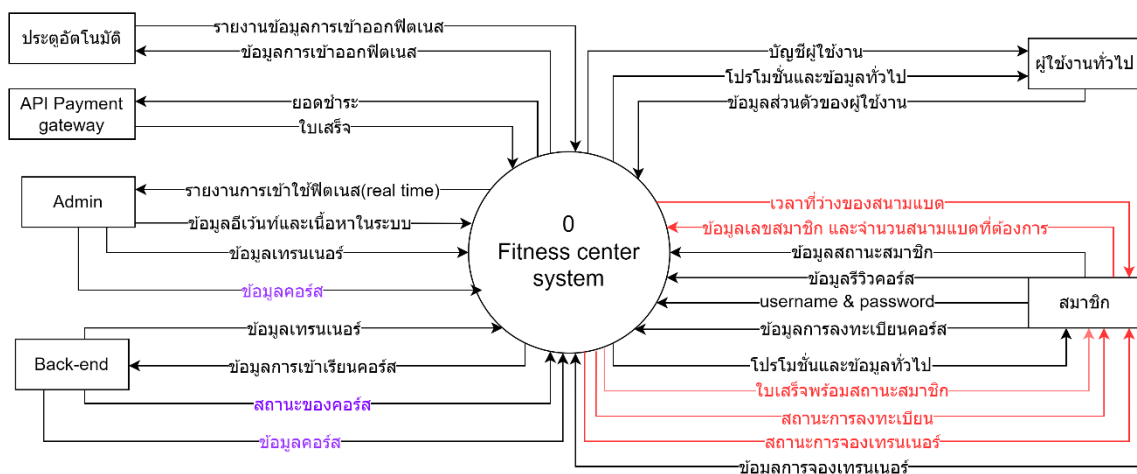


(Figure1.0 UseCaseDiagram:

<https://drive.google.com/file/d/1iK7g3i8TpDLNZljWwGkOLZIOEI4pOJka/view?usp=sharing>)

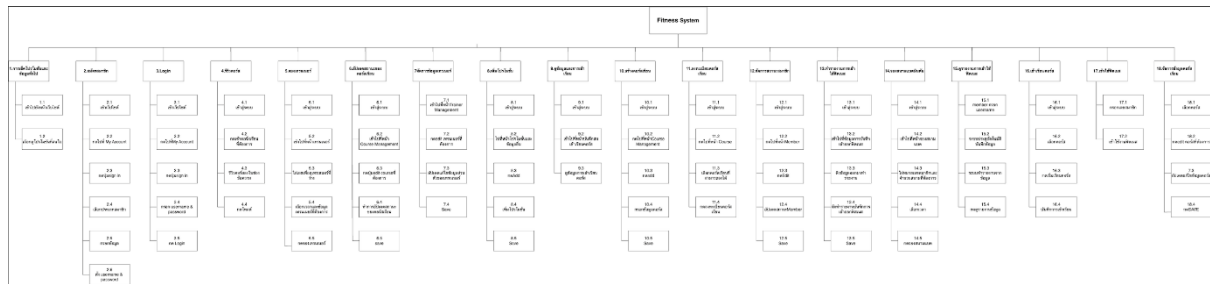
- การลดจำนวน Actor ทางฝั่งผู้ใช้งานลง และแยกประเภทสมาชิกด้วย use case ชำระเงินและไม่ชำระเงิน แทน
- การเชื่อม extend 2 use case(ชำระเงิน,ไม่ชำระเงิน) เข้าที่ use case ของสมัครสมาชิก และให้use case อื่นๆบางตัวโยง include เข้าที่ 2 use case นี้ด้วย เพื่อจำแนกว่า use case ไหนต้องชำระเงินก่อนจึงจะทำได้
- การเพิ่ม use case จองสนามแบดมินตันของ actor สมาชิก
- ประตุอัตโนมัติมีการเพิ่มuse case คือการบันทึกการเข้าออก เพราะ ประตุอัตโนมัติจะต้องมีการรวบรวมเก็บข้อมูลเพื่อใช้ในการบันทึก
- การเพิ่มuse caseดูบันทึกการเข้าออกของAdmin เพราะAdminสามารถมีการดูบันทึกการเข้าออกได้ทำให้actorระหว่างAdminและประตุอัตโนมัติจะไม่มีการเชื่อมต่อกันเพราะประตุอัตโนมัติมีuse caseที่บันทึกการเข้าออกเองแล้ว
- การตัดuse caseดูจำนวนผู้ใช้งานฟิตเนสของAdmin เพราะ มีการเพิ่มuse case บันทึกการเข้าออก
- การเปลี่ยนuse case ของ Adminส่วนของการจัดการข้อมูลและตีพิมพ์เนื้อหาเป็นการเพิ่มPromotion เพื่อให้ความถูกต้องและตรงกับRequirements
- แก้ไขการที่Back-endมีการสร้างคอร์สเรียนโดยให้การจัดการข้อมูลเป็นการextendเข้าไปเพราะว่าBack-endจะไม่จัดการข้อมูลของคอร์สเรียนหรือจัดการข้อมูลของคอร์สเรียนได้

2.แก้ไข Data Flow Diagram Level 0



- การแยกใบเสร็จพร้อมสถานะลงทะเบียนและใบเสร็จพร้อมข้อมูลการจองทรนเนอร์ออกจากกันเพราะว่าข้อมูลมีการเก็บที่แตกต่างกันทำให้ต้องออกมาเป็นใบเสร็จแต่ละใบเสร็จ
- มีการเปลี่ยนactor สมาชิกแบบฟรีและจ่ายเงินรวมเป็นสมาชิกอันเดียวเพราะมีการแก้ไขจาก ucd แล้ว ในdfdมีactorสมาชิกแบบเดียว
- มีการเพิ่มข้อมูลคอร์ส เข้าทั้ง Admin และ Back-end เพราะการจะสร้างคอร์สได้นั้นต้อง input ข้อมูลคอร์สเข้าไปก่อน
- มีการเพิ่ม input และ output ที่เกี่ยวข้องกับ use case การจองสนามแบดมินตันเข้าไปเพื่อให้สอดคล้องกัน

Functional Decomposition

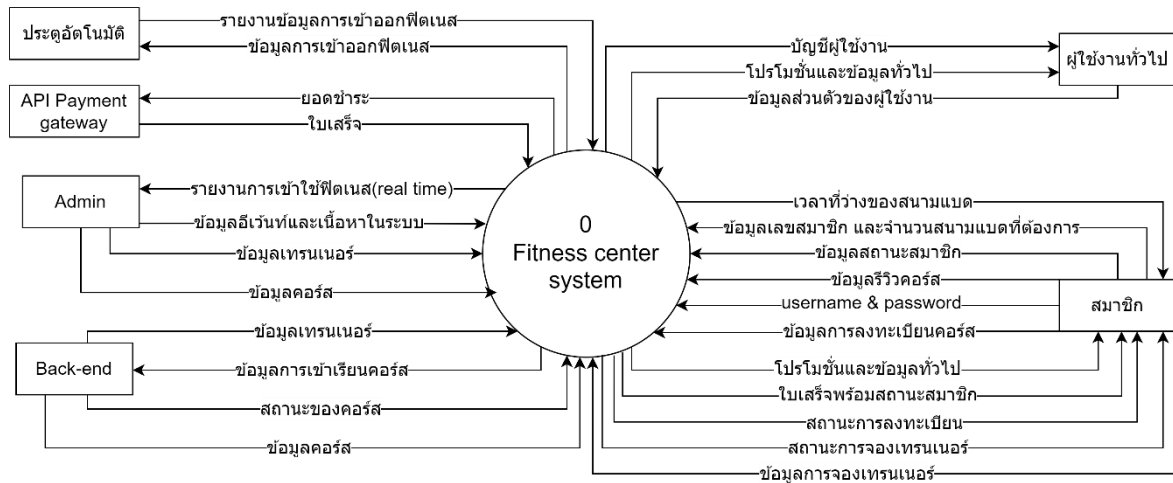


(Figure2.0 FunctionalDecomposition:

<https://drive.google.com/file/d/1iK7g3j8TpDLNZljWwGkOLZlQEI4pOJka/view?usp=sharing>)

DFD Diagram

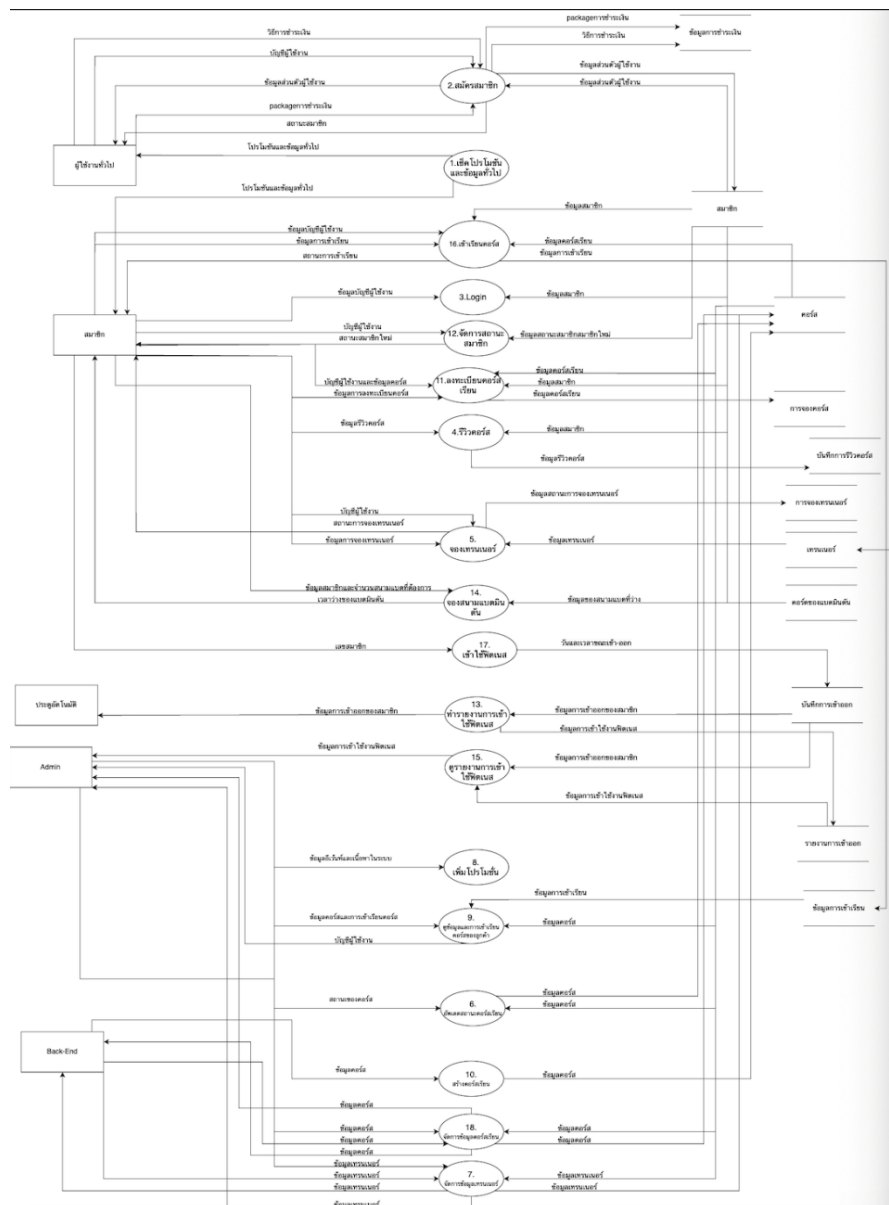
Data Flow Diagram Level 0 – context diagram



(Figure3.0 DFDLv.0:

<https://drive.google.com/file/d/1iK7g3j8TpDLNZljWwGkOLZIOEI4pOJka/view?usp=sharing>)

DFD level1



(Figure4.0 DFDLv.1:

<https://drive.google.com/file/d/1iK7g3j8TpDLNZljWwGkOLZlQEi4pOJka/view?usp=sharing>)

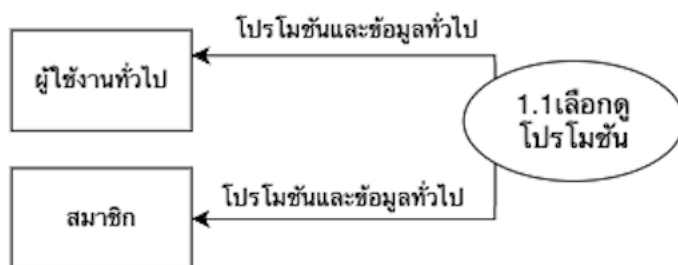
DFD Level2

(Figure5.0 DFDLv.2:

<https://drive.google.com/file/d/1iK7g3j8TpDLNZljWwGkOLZIOEI4pOJka/view?usp=sharing>)

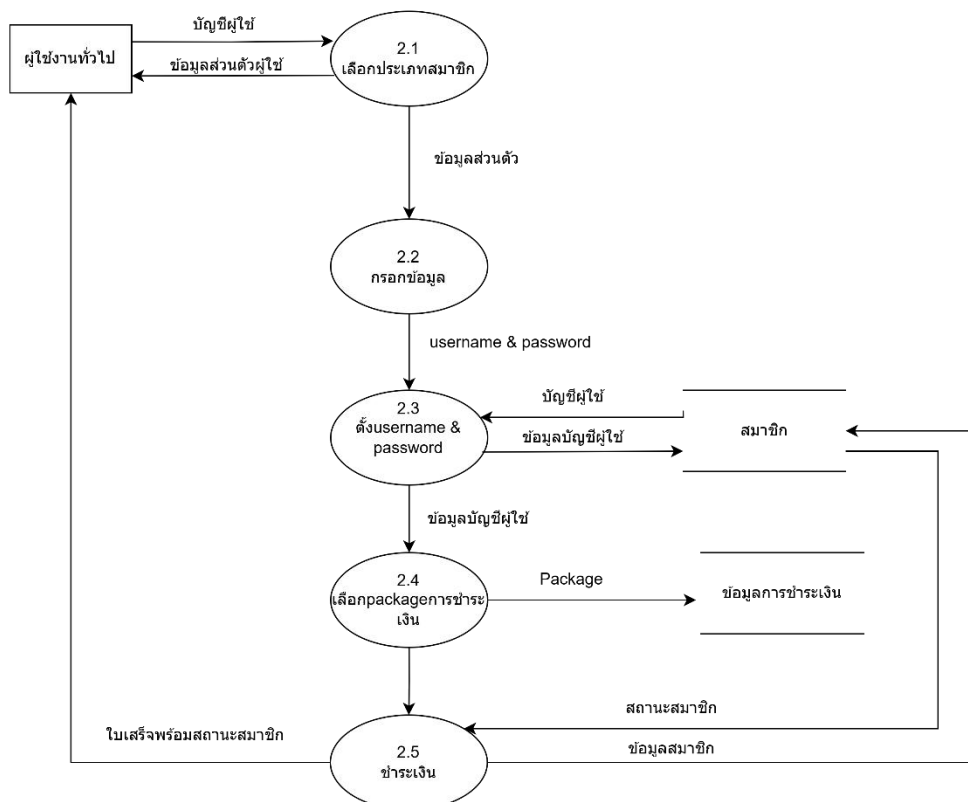
DFD Level 2 of process 1

กระบวนการที่1.0เช็คโปรโมชันและข้อมูลทั่วไป



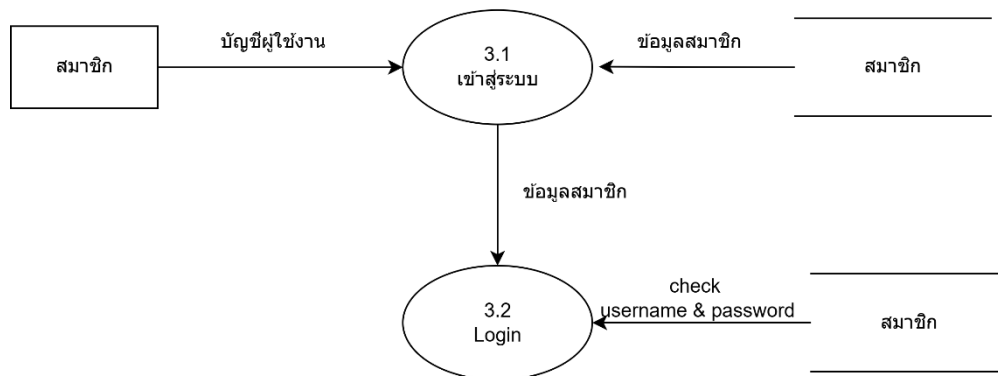
DFD Level 2 of process 2

กระบวนการที่2.0สมัครสมาชิก



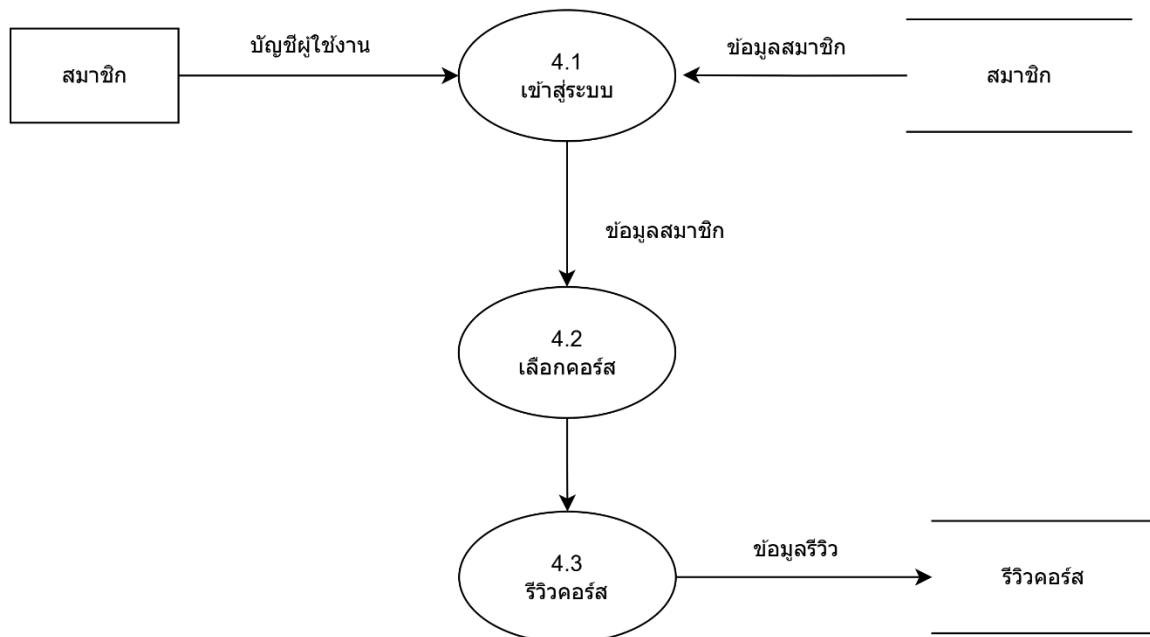
DFD Level 2 of process 3

กระบวนการที่3.0Login



DFD Level 2 of process 4

กระบวนการที่4.0รีวิวคอร์ส



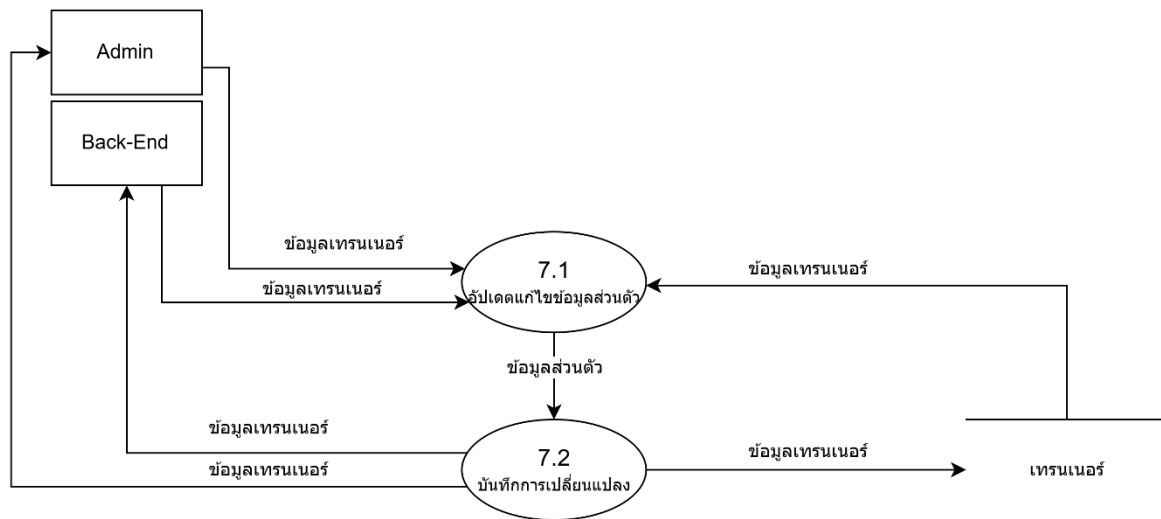
```
graph TD; User[สมาชิก] -- "บัญชีใช้งาน" --> S1((5.1 เข้าสู่ระบบ)); S1 -- "ข้อมูลสมาชิก" --> S2((5.2 เลือกเทนนิสที่ต้องการ)); S2 -- "ข้อมูลเทนนิส" --> S3((5.3 เลือกเวลาและข้อมูล)); S3 -- "ข้อมูลการจองเทนนิส" --> S4((5.4 กดยืนยันการจอง)); S4 -- "สถานะการจองเทนนิส" --> User; S4 -- "ข้อมูลการจองเทนนิส" --> User;
```

The flowchart illustrates the system process for selecting and booking a tennis court. It begins with a user (สมาชิก) who logs in (5.1 เข้าสู่ระบบ) using their account (บัญชีใช้งาน). The system then displays the available tennis courts (5.2 เลือกเทนนิสที่ต้องการ) based on the user's information (ข้อมูลสมาชิก). The user selects a court and time (5.3 เลือกเวลาและข้อมูล). The system then displays the booking status (5.4 กดยืนยันการจอง) based on the selected court and time (ข้อมูลการจองเทนนิส). The user confirms the booking (กดจองเทนนิส), and the system updates the booking status (สถานะการจองเทนนิส) and provides the booking information (ข้อมูลการจองเทนนิส) back to the user.

```
graph TD; BE[Back-End] -- "ข้อมูลสินค้า" --> P61((6.1)); BE -- "สถานะของคอร์ส" --> P61; C[คอร์ส] -- "ข้อมูลคอร์ส" --> P61; P61 -- "ข้อมูลคอร์ส" --> P62((6.2)); P62 -- "ข้อมูลคอร์ส" --> C; P61 --- L61[ทำการอัปเดตสถานะของ<br/>คอร์สเรียน]; P62 --- L62[บันทึกการเปลี่ยนแปลง];
```

DFD level 2 of process 7

กระบวนการที่ 7.0 จัดการข้อมูลเทรนเนอร์



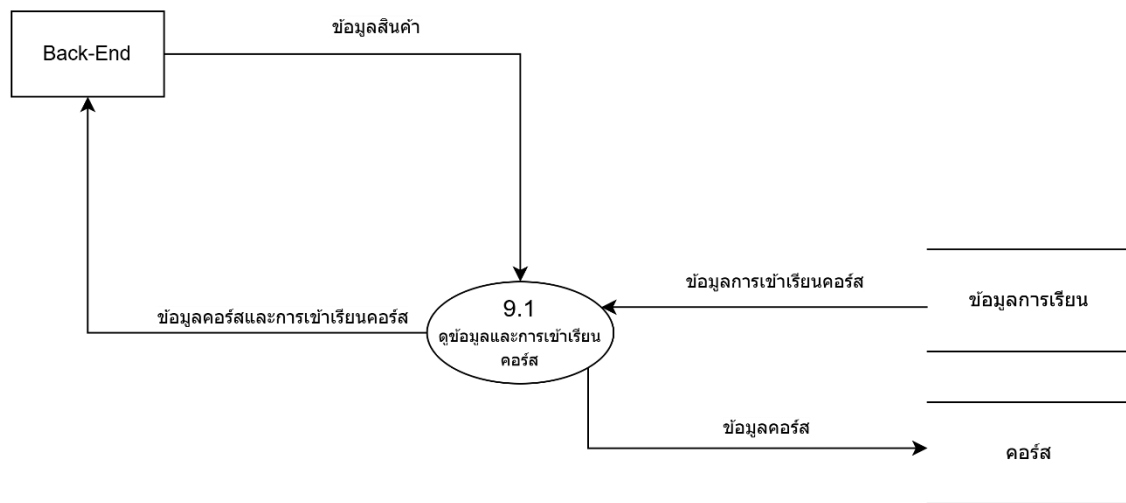
DFD Level 2 of process 8

กระบวนการที่ 8.0 เพิ่มโปรโมชั่น



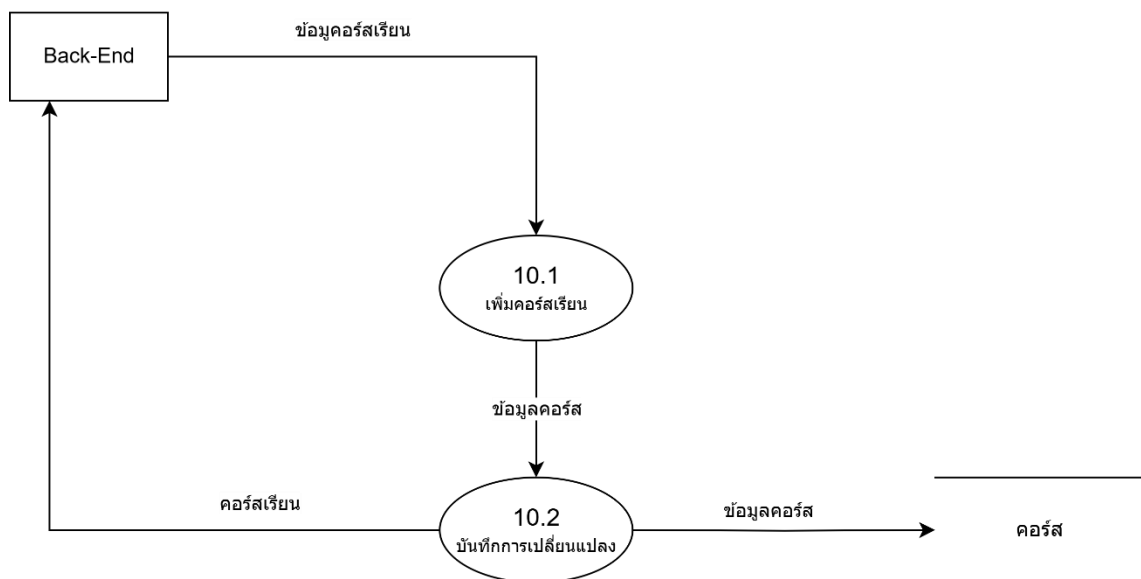
DFD Level 2 of process 9

กระบวนการที่ 9.0 ดูข้อมูลและการเข้าเรียน



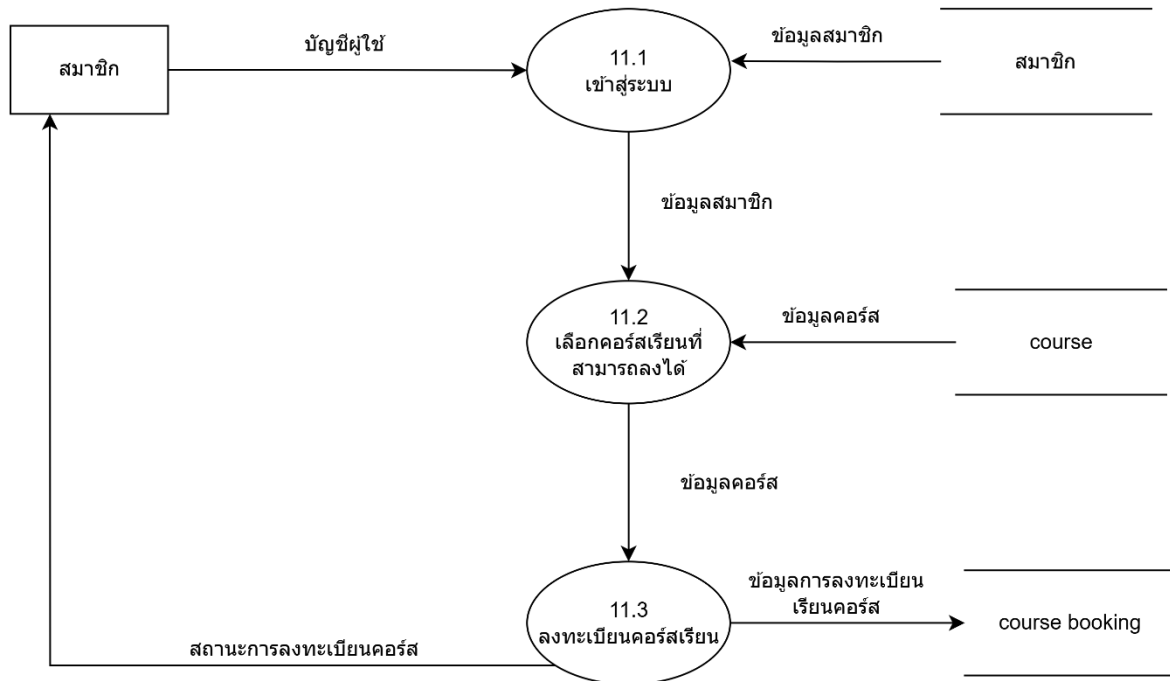
DFD Level 2 of process 10

กระบวนการที่ 10.0 สร้างคอร์สเรียน



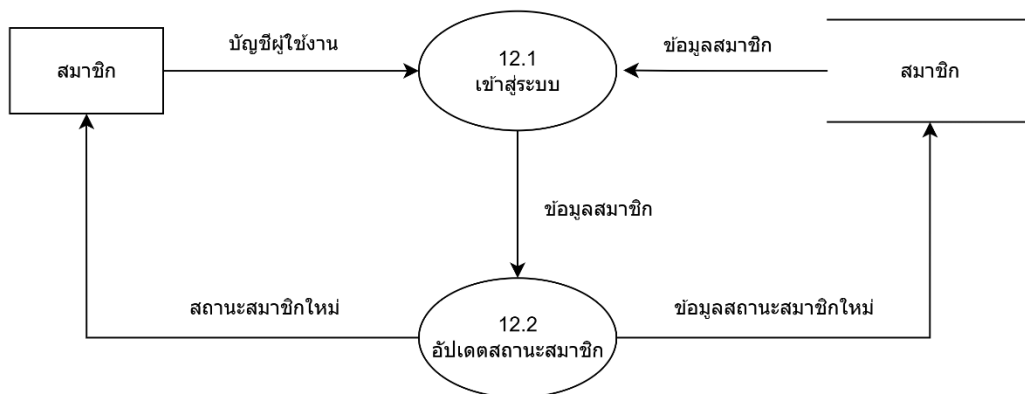
DFD Level 2 of process 11

กระบวนการที่11.0ลงทะเบียนคอร์สเรียน



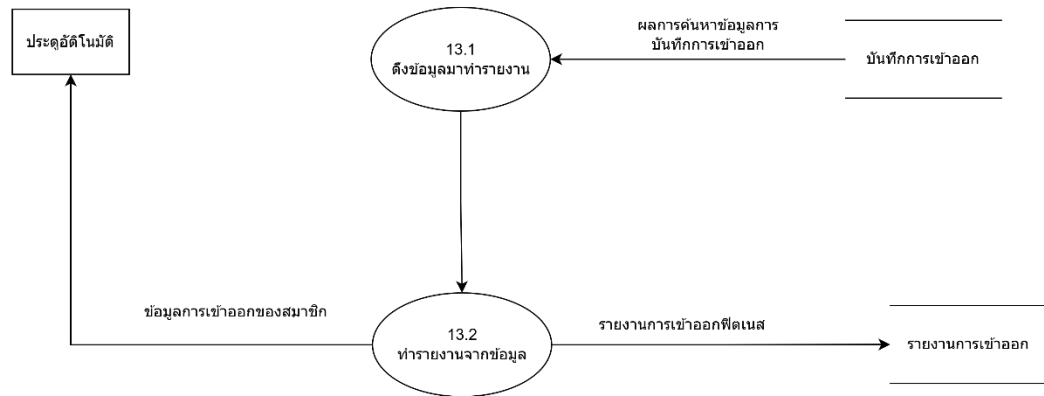
DFD Level 2 of process 12

กระบวนการที่12.0จัดการสถานะสมาชิก



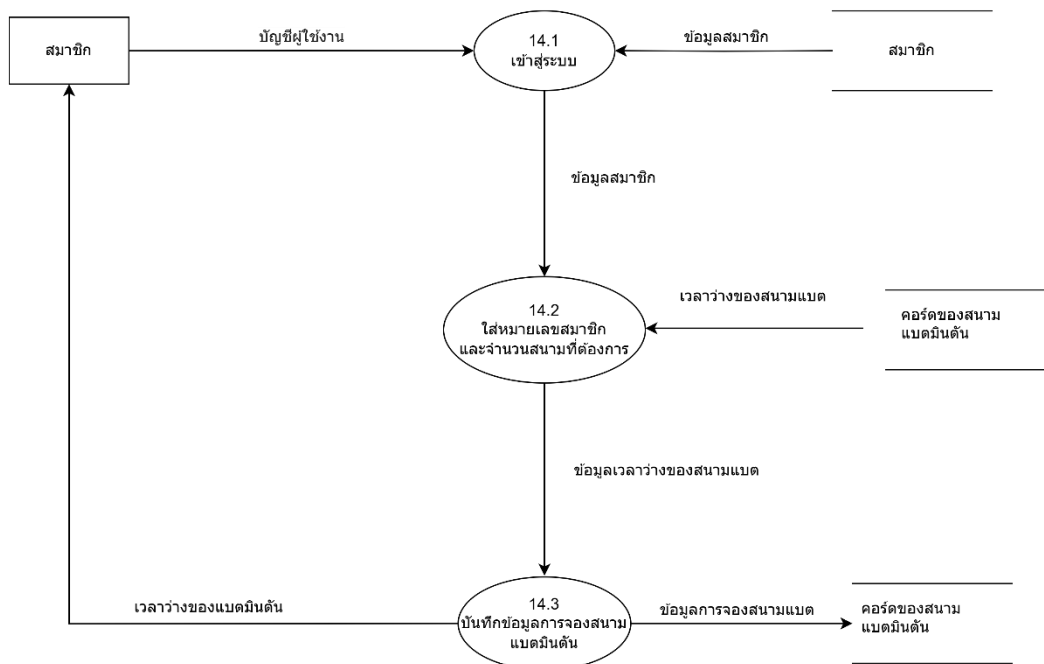
DFD Level 2 of process 13

กระบวนการที่ 13.0 ทำรายงานจากข้อมูล



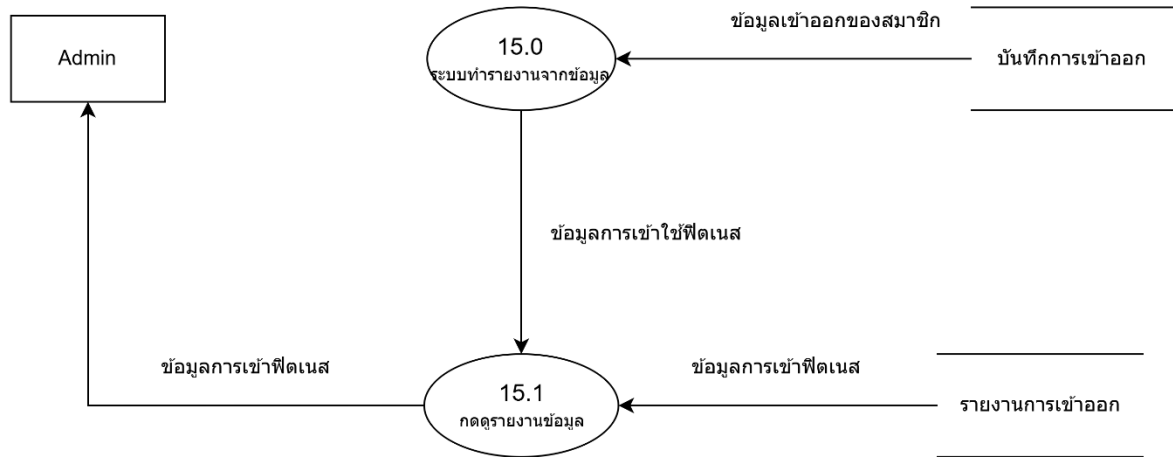
DFD Level 2 of process 14

กระบวนการที่ 14.0 จองสนามแบดมินตัน



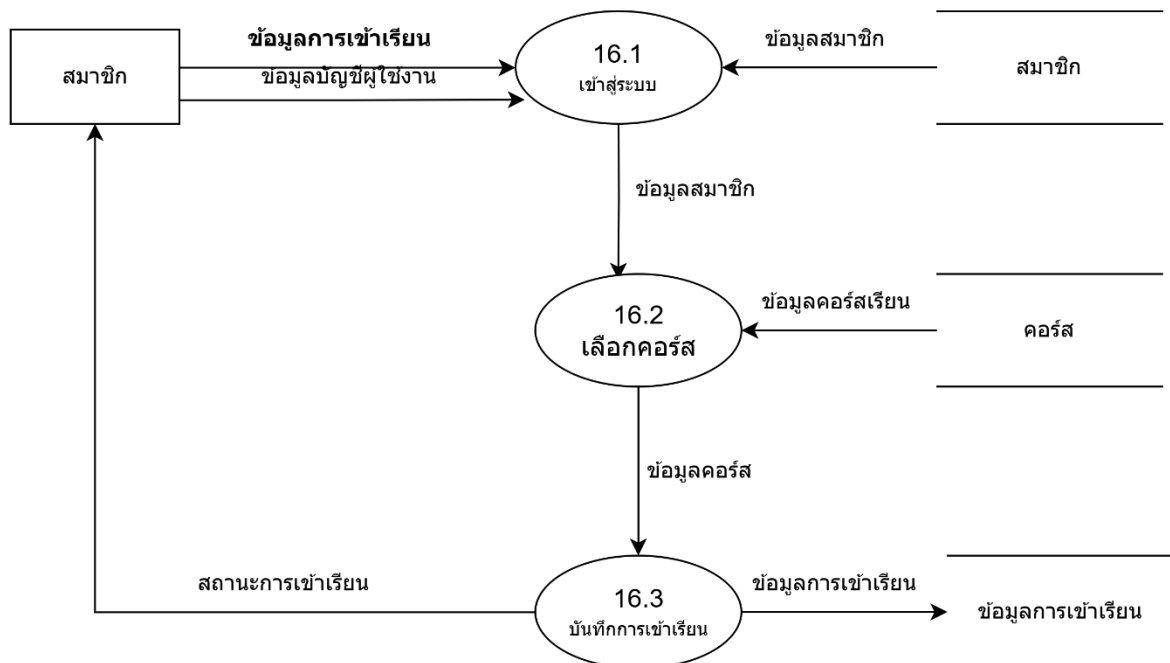
DFD Level 2 of process 15

กระบวนการที่ 15.0 ดูรายงานการเข้าใช้ฟีดเนส



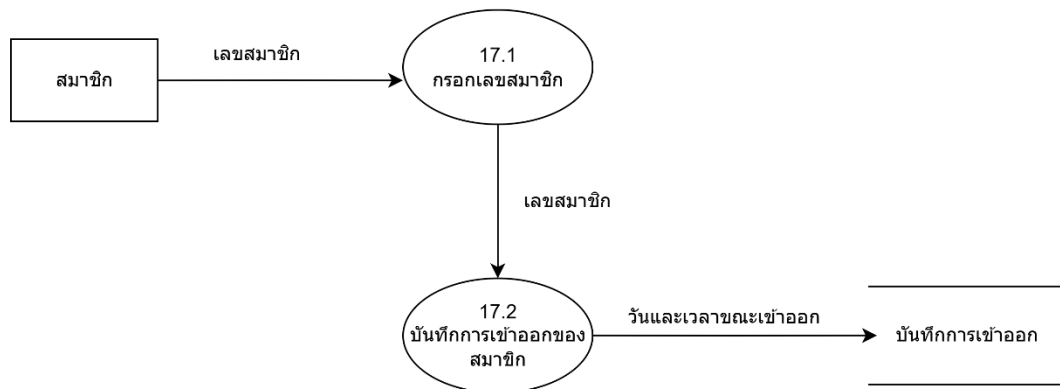
DFD Level 2 of process 16

กระบวนการที่ 16.0 เข้าคอร์สเรียน



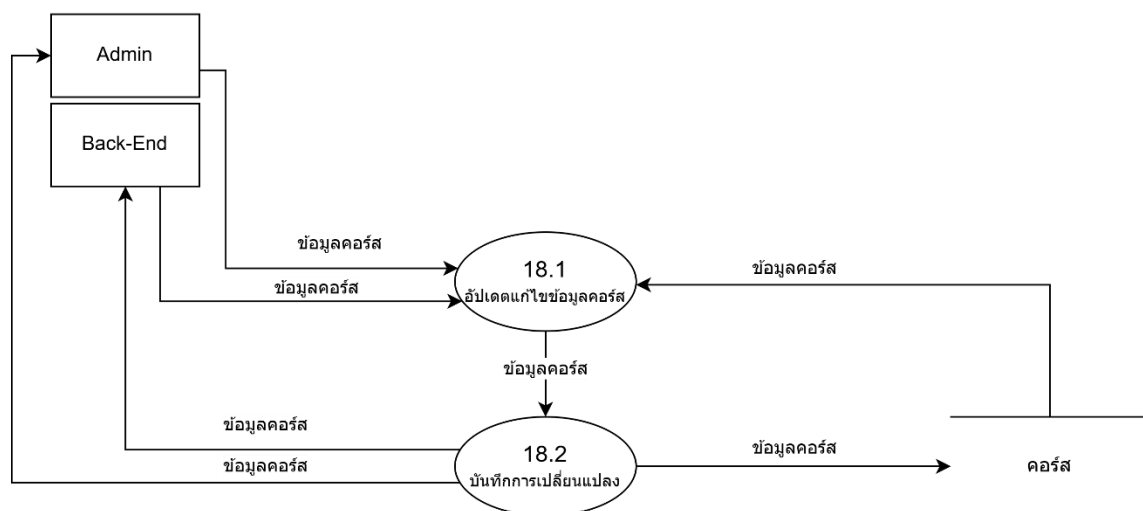
DFD Level 2 of process 17

กระบวนการที่ 17.0 เข้าใช้ฟีดแบ็ค



DFD Level 2 of process 18

กระบวนการที่ 18.0 จัดการข้อมูลคอร์สเรียน



List of data stores and attributes

1. ข้อมูลการชำระเงิน

- package ชำระเงิน
- วิธีชำระเงิน

2. สมาชิก

- ชื่อ
- วันเกิด
- อายุ
- ระดับความฟิต
- username
- password
- สถานะสมาชิก

3. คอร์ส

- ชื่อคอร์ส
- ชื่อผู้สอน
- ชนิดของคอร์ส
- กำหนดการของคอร์ส
- ระดับความฟิตขั้นต่ำ
- จำนวนนักเรียนที่รับได้

4. การลงทะเบียนคอร์ส

- ชื่อคอร์ส
- ชื่อสมาชิกที่ทำการลบ.
- วันและเวลาที่ลบ.ไว้

5. บันทึกการรีวิวคอร์ส

- ชื่อคอร์สที่ถูกรีวิว
- วันและเวลาที่รีวิว
- ข้อมูลที่สมาชิกทำการรีวิว

6. เทรนเนอร์

- ชื่อ
- วันเกิด
- อายุ
- เบอร์โทร
- อีเมลล์
- ความถนัดของเทรนเนอร์
- ระยะเวลาที่เทรนเนอร์ว่าง

7. การจองเทรนเนอร์

- ชื่อสมาชิกที่ทำการจอง
- ชื่อเทรนเนอร์ที่ถูกจอง
- วันและเวลาที่จอง

8. คอร์ดของแบดมินตัน

- หมายเลขคอร์ดสนามแบดมินตัน
- ระยะเวลาที่ใช้งานได้ของสนามแบดมินตัน
- สถานะของสนามแบดมินตัน

9. ข้อมูลการเข้าออกฟิตเนส

- เวลา ณ ขณะที่ผู้ใช้งานเข้าใช้ฟิตเนส
- เวลา ณ ขณะที่ผู้ใช้งานออกจากฟิตเนส

10. รายงานการเข้าออก

- จำนวนผู้ใช้งานฟิตเนส ณ ขณะนั้น
- จำนวนผู้ใช้งานฟิตเนสในแต่ละช่วงเวลา
- เวลาเข้า-ออกฟิตเนสของสมาชิกแต่ละคน

11. ข้อมูลการเข้าเรียน

- ชื่อคอร์ส
- ชื่อผู้ใช้ที่เข้าเรียน
- ระยะเวลาที่เข้าเรียน

Test Cases

Project Name : Fitness Center System

Test case ID : FIT_01

Test Priority : High

Module Name : Register for membership

Test Title : Try registering for free membership grade

Description : Test the page of free membership register form

Test designed by :

Pattaraporn

Test designed date :

18/11/2024

Test executed by :

Pattaraporn

Test executed date :

18/11/2024

Pre-condition : User have all the required information

Step	Test Steps	Test Data	Expected Result	Actual Result	Status
1	Navigate to the homepage				pass
2	Click the membership button				pass
3	Click the free membership button				pass
4	Provide name	Name: D			pass
5	Provide birthdate and click anywhere in the page	Birthdate: 06/06/2005	Age box should be 19	Age is 19	pass
6	Select the level	Level: LowFitness			pass
7	Provide username	Username: D			Pass
8	Provide password	Password: 123			pass
9	Click save		ManagementFree Mem page should	ManagementFr eeMem page	pass

			have the account D	should have account D	
--	--	--	-----------------------	--------------------------	--

Post-condition : User successfully creates a free membership account. The account details are kept in database.

Project Name : Fitness Center System

Test case ID : FIT_02

Test Priority : High

Module Name : Course Management

Test Title : Verify course creation with valid inputs

Description : Test the create course functionality in the course management

Test designed by :

Thitaree

Test designed date :

18/11/2024

Test executed by :

Thitaree

Test executed date :

18/11/2024

Pre-condition : Course data should be prepared for input.

Step	Test Steps	Test Data	Expected Result	Actual Result	Status
1	Navigate to the Course Management		User should be able to enter the Course Management page	User is navigated to the Course Management page	pass
2	Click the New Course button		User should be able to enter the page of new course form	User is navigated to the new course form page	pass
3	Enter a valid course name	Name: Yoga	Name field accepts the input.		

4	Enter a valid course description	Description: This course will help improve flexibility	Description field accepts the input.		
5	Select the type of course	Type: Flexibility	Type dropdown and field update correctly.		
6	Enter a valid course VO2max	VO2max: 35.00	VO2max field accepts the input.		
7	Enter a valid course start and end date	Schedule: 11/20/2024 - 11/30/2024	Date is accepted in the correct format (MM-DD-YYT).		
8	Select the trainer	Trainer: A a	Trainer dropdown and field update correctly		
9	Click the Save button		Course should be created.	Course is successfully created.	Pass

Post-condition : 1.New course is added to the system and visible in the course list. 2.Course details are logged in the database. 3.User can check and clear database about that course

Test Report

Test Cycle Fitness Center System

Executed pass : 2 fail : 0

Function	Description	%TCs Executed	%TCs Passed	TCs Pending	Priority
New Free Membership	Check new free membership account is created	100%	100%	0	High
New Course	Check new course is added	100%	100%	0	High