軟體工程開發實務 第16組 需求規格

以自然語言撰寫需求規格 (加選課程)

▪ 學生可以進行加選

▪ 學生不可選超過25學分

▪ 同一時段不可以有兩堂課程

▪ 彈出加選結果(成功/失敗)

以結構化的自然語言撰寫規格

▪ 格式化需求規格

▪ 功能定義

▪ 輸入與輸出

▪ 需要執行的動作

▪ 前置與後置條件 (Pre- & post-condition)

▪ 可能的副作用

以格式化自然語言描述需求規格

* 功能: 加選課程(以學生為例)
* 輸入: 學生的帳號、學生的密碼、學生待選的課程
* 輸出: 加選成功or失敗的訊息、課程餘額的變化、學生課表加入該課程(if 成功加選)
* 前置條件: 課程餘額>0
* 後置條件: 若加選成功，課程餘額-=1

若加選失敗，課程餘額不變

* 動作:

1. 系統檢查是否為加選時間
2. 使用者輸入自己的帳號
3. 使用者輸入自己的密碼
4. 系統進行查核身分
5. 使用者輸入自己的預選課程(只能一堂)
6. 系統進行查核加選後是否已達25學分
7. 系統進行查核是否衝堂
8. 系統顯示該課程可進行加選
9. 仍有餘額則可加選
10. 系統再次比對名額，若餘額>=1則選上
11. 該課程的課程餘額-1
12. 顯示加選成功
13. 將該課程的時間列於學生課表

替代流程:

1. 於基本流程第4步，查驗帳密後為錯
   1. 顯示
   2. 學生須將其他課程進行退選以達到條件
2. 於基本流程第6步，加選後已多於25學分
   1. 顯示學分數大於25學分
   2. 學生須將其他課程進行退選以達到條件
3. 於基本流程第7步，加選後會衝堂
   1. 顯示衝堂科目
   2. 顯示加選失敗
4. 於基本流程第10步，若點選時已無餘額
   1. 顯示該課已無名額
   2. 顯示加選失敗

----------------------------------------------------------------------

測試案例(一)

學生加選

學生資訊

帳號:D1158730

密碼:123456789

預選課程: 1432 軟體工程開發實務 (3學分)

目前學分數:20學分

1. 系統檢查是否為加選時間
2. 學生輸入自己的帳號
   1. 帳號:D1158730
3. 學生輸入自己的密碼
   1. 密碼:123456789
4. 系統進行查核身分
   1. 帳號驗證 - 有這個人
   2. 密碼驗證 - 正確
5. 學生輸入自己的預選課程
   1. 預選課程:1432
6. 系統進行查核加選後是否已達25學分
   1. 20+3=23
   2. <=25
7. 系統進行查核是否衝堂
   1. 無衝堂
8. 系統顯示該課程可進行加選
9. 仍有餘額則可加選
   1. 剩3名
10. 系統再次比對名額，若餘額>=1則選上
11. 該課程的課程餘額-1
    1. 剩2名
12. 顯示加選成功
13. 將該課程的時間列於學生課表

------------------------------------------------------------------------------------

測試案例(二)

學生加選

學生資訊

帳號:D1158730

密碼:123456789

預選課程: 1432 軟體工程開發實務 (3學分)

目前學分數:20學分

1. 系統檢查是否為加選時間
2. 使用者輸入自己的帳號
   1. 帳號:D1158730
3. 使用者輸入自己的密碼
   1. 密碼: ABCDEFGHJ
4. 系統進行查核身分
   1. 帳號驗證－有這個人
   2. 密碼驗證－錯了
   3. 回到第２步

------------------------------------------------------------------------------------

測試案例(三)

學生加選

學生資訊:

帳號:D1158730

密碼:123456789

預選課程: 1432 軟體工程開發實務 (3學分)

目前學分數:23學分

1. 系統檢查是否為加選時間
2. 使用者輸入自己的帳號
   1. 帳號:D1158730
3. 使用者輸入自己的密碼
   1. 密碼:123456789
4. 系統進行查核身分
   1. 帳號驗證 - 有這個人
   2. 密碼驗證 - 正確
5. 使用者輸入自己的預選課程(只能一堂)
   1. 1432
6. 系統進行查核加選後是否已達25學分
   1. 23+3=26
   2. >25
   3. 顯示學分數超過25學分
   4. 回到第２步

------------------------------------------------------------------------------------

以格式化自然語言描述需求規格

* 功能: 開課
* 輸入: 系助的帳號、系助的密碼、課程資訊
* 輸出: 加入成功or失敗的訊息、課程資訊的變化、教授是否可以填寫大綱
* 前置條件: 同一時間該課教室及教授時間不衝突
* 後置條件: 若加入成功，課程+1

若加入失敗，無此課堂

* 動作:

1. 系統檢查是否為加入時間
2. 使用者輸入自己的帳號
3. 使用者輸入自己的密碼
4. 系統進行查核身分
5. 使用者輸入欲加入的課堂時間地點
6. 系統進行查核是否教室時間衝突
7. 使用者輸入欲加入的課堂授課教授
8. 系統顯示該教授時間是否衝突
9. 以上皆通過再進行
10. 使用者輸入欲加入的課堂名稱
11. 使用者輸入欲加入的課堂名額
12. 使用者輸入欲加入的課堂代號
13. 使用者輸入欲加入的課堂類型
14. 使用者點選加入"按鍵"
15. 系統輸入進資料庫
16. 系統回傳創建成功
17. 授課教授的課表會出現這堂課
18. 授課教授可以輸入授課大綱

替代流程:

1. 於基本流程第4步，查驗帳密後為錯
   1. 顯示
   2. 重新輸入帳密
2. 於基本流程第6步，教室時間衝突
   1. 顯示該教室已有課堂使用
   2. 取消輸入值
3. 於基本流程第8步，教授時間衝突
   1. 顯示教授該時間已有分配
   2. 取消輸入值

—---------------------------------------------------------------------------------

測試案例(一)

系助加選

系助資訊

帳號:IECSwu

密碼:KKKKKK

加入課程: 1432 軟體工程開發實務 (3學分)

時間地點:週三6~8節 資電330

1. 系統檢查是否為加選時間
2. 系助輸入自己的帳號
   1. 帳號:IECSwu
3. 系助輸入自己的密碼
   1. 密碼: KKKKKK
4. 系統進行查核身分
   1. 帳號驗證－有這個人
   2. 密碼驗證－正確
5. 系助輸入週三 6~8 資電330
6. 系統進行查核是否教室時間衝突
   1. 時間地點驗證 - 空堂
7. 系助輸入授課教授 - 許懷中
8. 系統顯示該教授時間是否衝突
   1. 教授時間驗證 - 空堂
   2. 顯示教授時間衝突
9. 使用者輸入欲加入的課堂名稱
10. 使用者輸入欲加入的課堂名額
11. 使用者輸入欲加入的課堂代號
12. 使用者輸入欲加入的課堂類型
13. 使用者點選加入"按鍵"
14. 系統輸入進資料庫
15. 系統回傳創建成功
16. 授課教授(許懷中)的課表會出現這堂課
17. 授課教授(許懷中)可以輸入授課大綱

------------------------------------------------------------------------------------

測試案例(二)

系助加選

系助資訊

帳號:IECSwu

密碼:KKKKKK

加入課程: 1432 軟體工程開發實務 (3學分)

時間:週三6~8節 資電125 (當作地點時間衝突)

1. 系統檢查是否為加選時間
2. 系助輸入自己的帳號
   1. 帳號:IECSwu
3. 系助輸入自己的密碼
   1. 密碼: KKKKKK
4. 系統進行查核身分
   1. 帳號驗證－有這個人
   2. 密碼驗證－正確
5. 系助輸入週三 6~8 資電330
6. 系統進行查核是否教室時間衝突
   1. 時間地點驗證 - 有課
   2. 顯示該教室已有課程
   3. 退回第5步

------------------------------------------------------------------------------------

測試案例(三)

系助加選

系助資訊

帳號:IECSwu

密碼:KKKKKK

加入課程: 1432 軟體工程開發實務 (3學分)

時間:週三1~3節 資電330 (教師時間衝突)

系統檢查是否為加選時間

系助輸入自己的帳號

帳號:IECSwu

系助輸入自己的密碼

密碼: KKKKKK

系統進行查核身分

帳號驗證－有這個人

密碼驗證－正確

系助輸入週三 6~8 資電330

系統進行查核是否教室時間衝突

時間地點驗證 - 空堂

系助輸入授課教授 - 許懷中

系統進行空堂查驗

系統顯示該教授時間衝突

退回第5步

------------------------------------------------------------------------------------

以格式化自然語言描述需求規格

* 功能: 退選(以學生為例)
* 輸入: 待退的課程
* 輸出: 退選成功or失敗or需進行後續處理的訊息、學生課表消除該課程(if 成功退選)
* 前置條件: 目前學分 - 待退的課程學分大於最低學分限制(9)
* 後置條件: 若退選成功，目前學分 =- 待退的課程學分

若退選失敗，課程餘額不變

若退選需進行後續處理，通知系助相關資訊

* 動作:

1. 系統檢查是否為退選時間
2. 使用者登入
3. 使用者選擇需退的課程(只能一堂)
4. 使用者執行退選
5. 系統進行查核身分
6. 系統進行查核退選後課程學分是否多於最低學分限制(9)
7. 系統進行查核是否必修課
8. 從該課程中刪除學生
9. 課程餘額+1
10. 使用者目前學分減去退選的課程學分
11. 系統顯示退選成功

* 替代流程:

1. 使用者執行退選(S4) - 使用者尚未選擇想要取消註冊的課程
   1. 系統提醒使用者先選擇課程
   2. 返回步驟 3
2. 系統進行查核退選後學分是否多於最低學分限制(S6) - 學分低於最低學分限制
   1. 系統提醒使用者目前的學分太低
   2. 返回步驟 3
3. 系統進行查核是否必修課(S7) - 需退的課程是必修課
   1. 系統記錄請求並將其狀態標記為“待處理”
   2. 通知系助有學生已申請退選必修課並需跟進
   3. 通知使用者退選需尋找系助跟進

—----------

測試案例(一)

學生資訊

目前學分:12

退選課程: 0001 工程數學 (3學分) – 非必修

1. 系統檢查是否為退選時間
2. 使用者登入
3. 使用者選擇線性代數
4. 使用者執行退選
5. 系統進行查核身分 -> 確認已註冊課程
6. 系統進行查核退選後課程學分是否多於最低學分限制(9) -> 12-3=9>=9 通過
7. 系統進行查核是否必修課 ->不是
8. 從該課程中刪除學生
9. 課程餘額+1
10. 使用者目前學分減去退選的課程學分
11. 系統顯示退選成功

測試案例(二)

學生資訊

目前學分:9

退選課程: 0001 工程數學 (3學分) – 非必修

1. 系統檢查是否為退選時間
2. 使用者登入
3. 使用者選擇線性代數
4. 使用者執行退選
5. 系統進行查核身分 -> 確認已註冊課程
6. 系統進行查核退選後課程學分是否多於最低學分限制(9) -> 9-3=6<9 不通過
7. 系統提醒使用者目前的學分太低
8. 退選失敗

—----------

測試案例(三)

學生資訊

目前學分:12

退選課程: 0002 線性代數 (3學分) - 必修

1. 系統檢查是否為退選時間
2. 使用者登入
3. 使用者選擇線性代數
4. 使用者執行退選
5. 系統進行查核身分 -> 確認已註冊課程
6. 系統進行查核退選後課程學分是否多於最低學分限制(9) -> 12-3=9 >=9 通過
7. 系統進行查核是否必修課 ->是
8. 系統記錄請求並將其狀態標記為“待處理”
9. 通知系助有學生已申請退選必修課並需跟進
10. 通知使用者退選需尋找系助跟進

**use case diagram**

連結: <https://drive.google.com/file/d/1my2fJt4-vx4dbZbO0bVfxuOhPgZme4eu/view?usp=sharing>

