

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN I



MÔN HỌC: NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Nhóm BTL:	10
Tên BTL:	Hệ thống quản lí lịch thi đấu và kết quả giải đua công thức 1 F1
Danh sách thành viên:	1. Trần Văn Duy – B22DCCN159 2. Vũ Trọng Khôi – B22DCCN468 3. Nguyễn Đức Mạnh - B22DCKH075 4. Hoàng Phúc Đức Minh - B22DCCN531
Yêu cầu:	Báo cáo tổng hợp
Modul:	Cập nhật kết quả

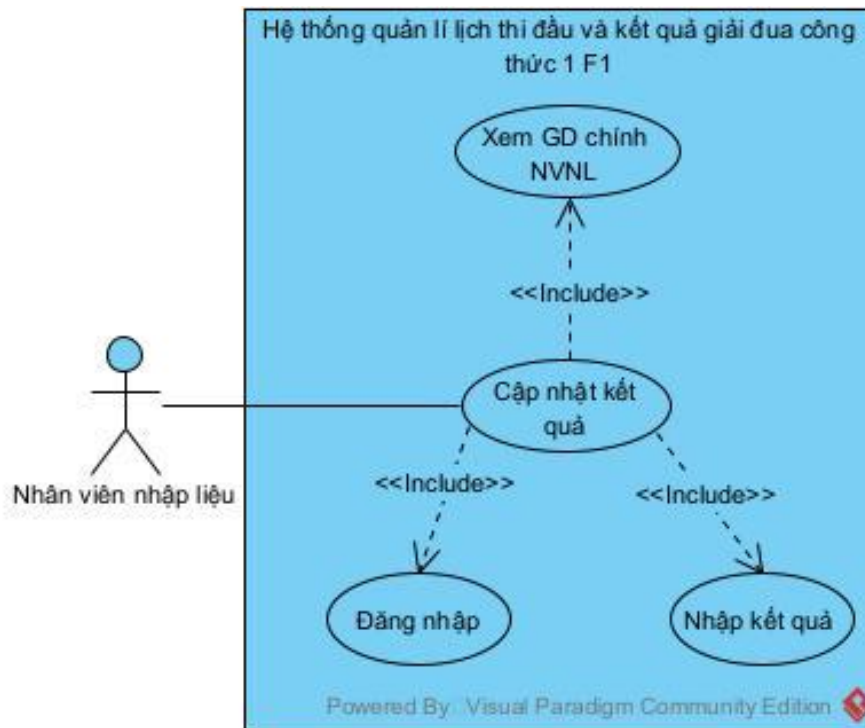
Hà Nội – 2025

Mục lục

1. Biểu đồ UC chi tiết của modul + mô tả các UC của modul	2
a. Usecase chi tiết	2
b. Mô tả Usecase chi tiết.....	2
2. Kịch bản chuẩn của modul	2
3. Biểu đồ thực thể pha phân tích của modula.....	5
a. Diễn giải.....	5
b. Biểu đồ lớp thực thể	7
4. Biểu đồ lớp đầy đủ pha phân tích của modul	7
a. Diễn giải.....	7
b. Biểu đồ lớp đầy đủ.....	8
5. Biểu đồ tuần tự pha phân tích của modul	8
a. Diễn giải kịch bản chuẩn v2	8
b. Biểu đồ tuần tự pha phân tích.....	10
6. Biểu đồ thiết kế lớp thực thể của modul.....	10
a. Diễn giải.....	10
b. Biểu đồ lớp thực thể	12
7. Biểu đồ thiết kế CSDL của modul.....	12
a. Diễn giải.....	12
b. Kết quả biểu đồ thiết kế CSDL.....	13
8. TK giao diện và biểu đồ lớp thiết kế chi tiết đầy đủ của modul.....	13
a. Thiết kế giao diện	13
b. Biểu đồ lớp thiết kế chi tiết đầy đủ.....	17
9. Biểu đồ tuần tự pha thiết kế của modul	18
a. Diễn giải kịch bản chuẩn v3	18
b. Biểu đồ tuần tự pha thiết kế.....	20
10. Test plan và test case chuẩn cho test hộp đen của modul	21
a. Blackbox test plan.....	21
b. Testcase Chuẩn	21

1. Biểu đồ UC chi tiết của modul + mô tả các UC của modul

a. Usecase chi tiết



b. Mô tả Usecase chi tiết

- UC "Đăng nhập" cho phép UC chính "Cập nhật kết quả" xác thực tài khoản của người dùng.
- UC "Xem giao diện chính NVNL" cho phép UC chính "Cập nhật kết quả" hiển thị giao diện chính của NVNL cho người dùng.
- UC "Nhập kết quả" cho phép UC chính "Cập nhật kết quả" thực hiện cập nhật kết quả của các tay đua.

2. Kịch bản chuẩn của modul

Scenario	Cập nhật kết quả.
Actors	Nhân viên nhập liệu (NVNL).
Pre-condition	NVNL đăng nhập thành công.
Post-condition	Kết quả thi đấu của các tay đua được cập nhật.

Main events

1. NVNL A đăng nhập vào hệ thống với username: a, password: a@123.

2. HT hiện GD chính của NVNL có lựa chọn Cập nhật kết quả.

3. NVNL chọn chức năng Cập nhật kết quả.

4. HT hiện GD chức năng Cập nhật kết quả gồm:

- ComboBox chọn chặng đua.

Chọn chặng đua

- Danh sách tay đua dạng bảng (chưa có dữ liệu) gồm các cột: STT, Tên, Tên đội đua, Quốc tịch, Ngày sinh, Tiểu sử, Thời gian về đích, Số vòng đua hoàn thành.
(Thời gian về đích và Số vòng đua hoàn thành là dạng ô nhập)

STT	Tên	Tên đội đua	Quốc tịch	Ngày sinh	Tiểu sử	Thời gian về đích	Số vòng đua hoàn thành

- Nút "Lưu".

5. NVNL click vào ComboBox.

6. HT hiển thị danh sách xổ xuống tên của các chặng đua đã kết thúc.

Chọn chặng đua

Hà Nội

Đà Nẵng

TP HCM

7. NVNL click chọn chặng đua "Hà Nội" từ danh sách xổ xuống tên các chặng đua.

8. HT hiện ra danh sách các tay đua đã đăng kí thi đấu cho chặng đua "Hà Nội":

STT	Tên	Tên đội đua	Quốc tịch	Ngày sinh	Tiểu sử	Thời gian về đích	Số vòng đua hoàn thành
-----	-----	-------------	-----------	-----------	---------	-------------------	------------------------

	1	Nguyễn Văn A	ABC	Việt Nam	21/10/1999	Không có		
	2	Nguyễn Văn B	ABC	Việt Nam	15/01/1998	Không có		
	3	Lê Văn C	XYZ	Việt Nam	10/01/1997	Không có		
	4	Lê Văn D	XYZ	Việt Nam	10/05/1997	Không có		
	9. NVNL lần lượt nhập đầy đủ Thời gian về đích và Số vòng đua hoàn thành của tất cả tay đua và click nút "Lưu".							
	STT	Tên	Tên đội đua	Quốc tịch	Ngày sinh	Tiêu sử	Thời gian về đích	Số vòng đua hoàn thành
	1	Nguyễn Văn A	ABC	Việt Nam	21/10/1999	Không có	1:33:10.021	53
	2	Nguyễn Văn B	ABC	Việt Nam	15/01/1998	Không có	1:32:52.121	53
	3	Lê Văn C	XYZ	Việt Nam	10/01/1997	Không có	1:33:14.021	53
	4	Lê Văn D	XYZ	Việt Nam	10/05/1997	Không có	1:32:45.321	53
10. HT kiểm tra tính hợp lệ của dữ liệu và lưu kết quả của các tay đua được nhập vào cơ sở dữ liệu. HT thông báo lưu kết quả thành công.								
Exceptions	2. HT báo sai username/password 2.1. NVNL click nút OK của thông báo. 2.2. HT hiện lại GD đăng nhập với username = a, password = a@123. 2.3. NVNL sửa lại password = a#123 và click Đăng nhập. 2.4. HT hiện GD chính của NVNL 10.1. HT báo “Ô nhập không được để trống” 10.1.1. NVNL click nút OK của thông báo, điền đầy đủ dữ liệu các ô trống và click Lưu.							

	<p>10.1.2. HT kiểm tra tính hợp lệ của dữ liệu và lưu kết quả vào cơ sở dữ liệu. HT thông báo lưu kết quả thành công.</p> <p>10.2. HT báo “Dữ liệu nhập không hợp lệ”</p> <p>10.2.1. NVNL click nút OK của thông báo.</p> <p>10.2.2. HT xóa dữ liệu tại các ô nhập có dữ liệu không hợp lệ.</p> <p>10.2.3. NVNL nhập đầy đủ dữ liệu theo đúng định dạng và click Lưu.</p> <p>10.2.4. HT kiểm tra tính hợp lệ của dữ liệu và lưu kết quả vào cơ sở dữ liệu. HT thông báo lưu kết quả thành công.</p>
--	---

3. Biểu đồ thực thể pha phân tích của modula.

a. Diễn giải

* Trích, đánh giá danh từ:

STT	Danh từ	Đánh giá danh từ
1	Nhân viên nhập liệu	(1) Thực thể User: username, password, fullname, role.
2	Hệ thống	Loại: quá chung chung
3	Username/password	Thuộc tính
4	Giao diện	Loại: quá chung chung
5	Kết quả	Loại: quá chung chung
6	Chức năng	Loại: quá chung chung
7	ComboBox	Loại: quá chung chung
8	Danh sách	Loại: quá chung chung

9	Tên	Thuộc tính
10	Chặng đua	(2) Thực thể RaceTrack: name, numOfLaps, location, time, des.
11	Tay đua	(3) Thực thể Driver: name, dob, national, bio.
12	Đội đua	(4) Thực thể RaceTeam: name, brand, des.
13	Bảng	Loại: quá chung chung
14	Dữ liệu	Loại: quá chung chung
15	Số thứ tự	Loại: quá chung chung
16	Tên, Tên đội đua, Quốc tịch, Ngày sinh, Tiểu sử.	Thuộc tính
17	Số vòng đua hoàn thành, Thời gian hoàn thành.	Loại: Trừu tượng
18	Ô nhập	Loại: quá chung chung
19	Nút	Loại: quá chung chung
20	Cơ sở dữ liệu	Loại: quá chung chung

*** Các lớp thực thể:**

- Người dùng - User: username, password, fullName, role.
- Chặng đua - RaceTrack: name, numOfLaps, location, time, des.
- Tay đua - Driver: name, dob, national, bio.
- Đội đua - RaceTeam: name, brand, des.

*** Quan hệ giữa các lớp thực thể:**

- Một tay đua có thể tham gia nhiều đội đua ở những thời điểm khác nhau, một đội đua gồm nhiều tay đua.

→ Driver – RaceTeam là quan hệ n – n.

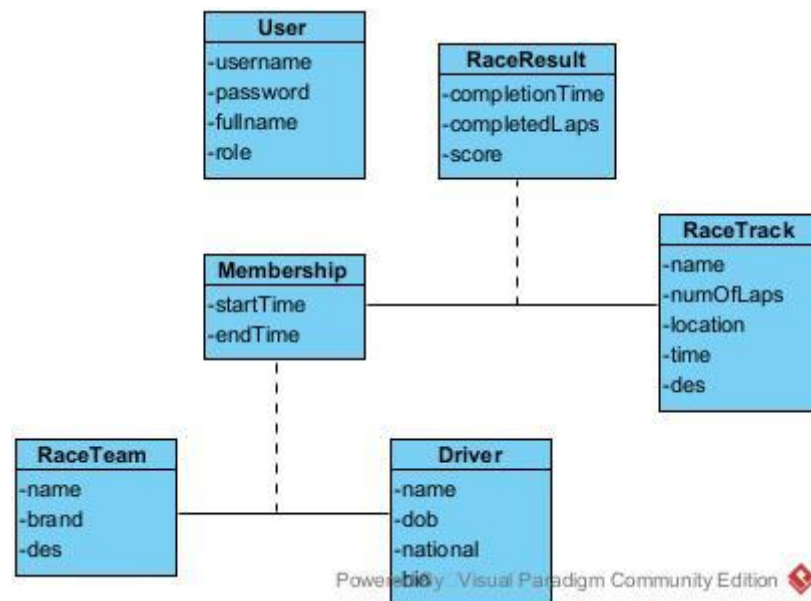
→ Đề xuất thực thể Thành viên - Memberships : mỗi Driver và RaceTeam sẽ xác định một Membership

- Mỗi tay đua khi tham gia một đội đua có thể thi đấu nhiều chặng đua khác nhau, mỗi chặng đua có nhiều tay đua tham gia.

→ Membership – RaceTrack là quan hệ n – n.

→ Đề xuất thực thể RaceResult (Kết quả thi đấu): mỗi Membership và RaceTrack sẽ xác định một RaceResult.

b. Biểu đồ thực thể



4. Biểu đồ lớp đầy đủ pha phân tích của modul

a. Diễn giải

* GD Đăng nhập:

Hàm kiểm tra đăng nhập

- Tên: checkLogin()
- Input: username, password (User)
- Output: boolean
- Lớp chủ thể: User

* GD Cập nhật kết quả:

Hàm lấy danh sách chặng đua:

- Tên: getRaceTrackList()
- Input: none

- Output: List<RaceTrack>
- Lớp chủ thể: RaceTrack

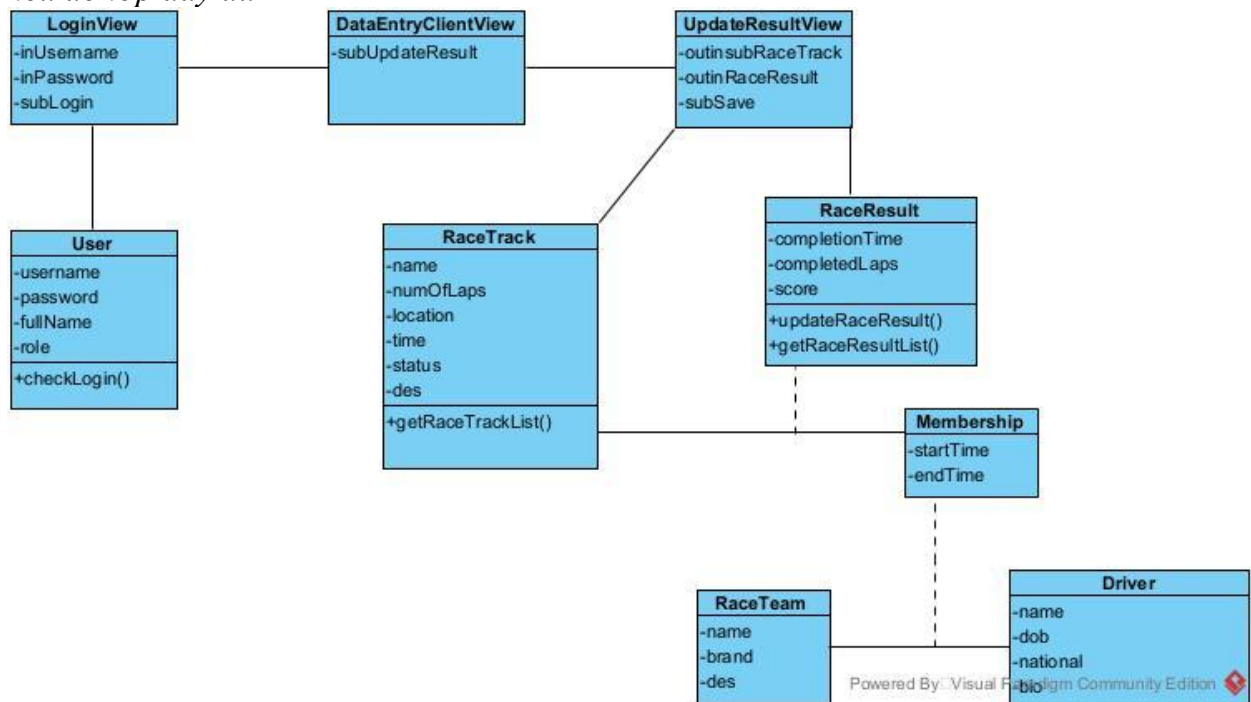
Hàm lấy danh sách các kết quả của chặng đua

- Tên: getRaceResultList()
- Input: idRaceTrack (RaceTrack)
- Output: List<RaceResult>
- Lớp chủ thể: RaceResult.

Hàm cập nhật kết quả thi đấu:

- Tên: updateRaceResult()
- Input: completionTime, completedLaps (RaceResult)
- Output: xác nhận thành công.
- Lớp chủ thể: RaceResult

b. Biểu đồ lớp đầy đủ



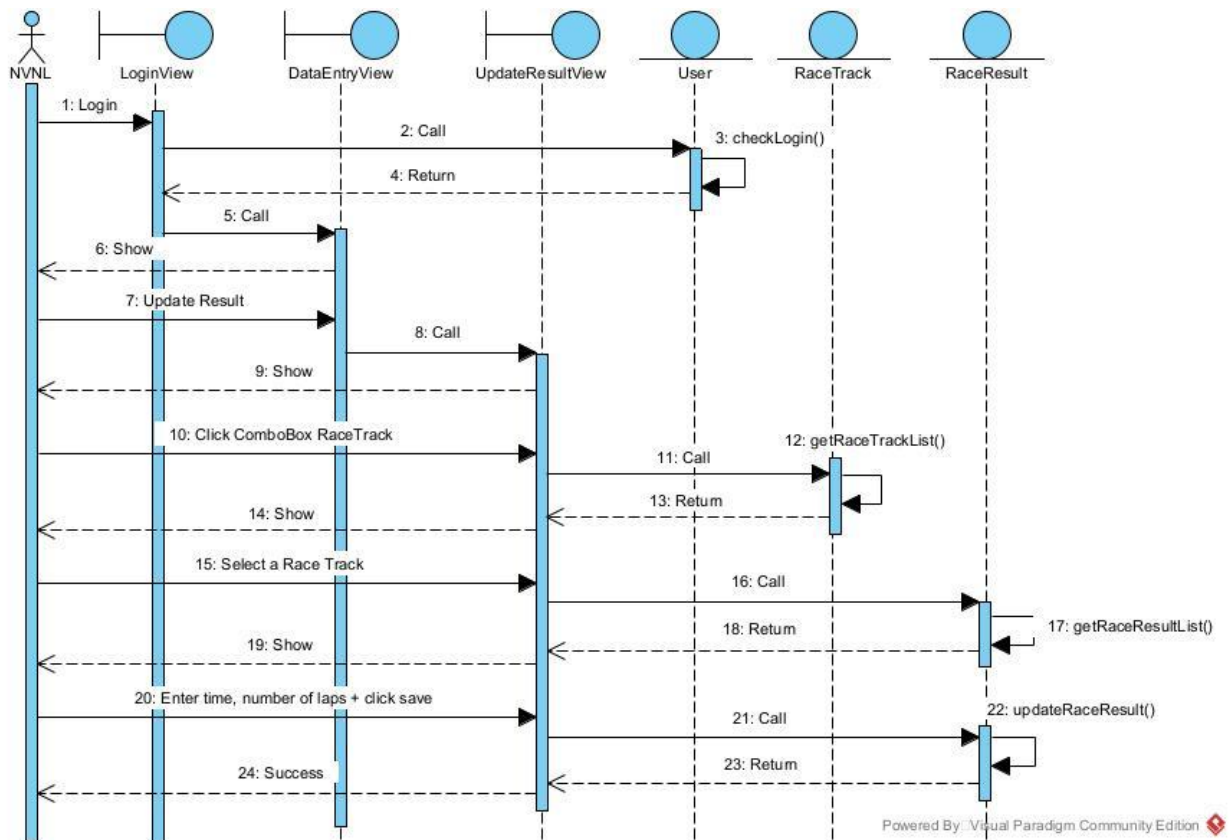
5. Biểu đồ tuần tự pha phân tích của modul

a. Diễn giải kịch bản chuẩn v2

1. NVNL đăng nhập trên giao diện LoginView.
2. LoginView gọi lớp User để xử lý.
3. Lớp User thực hiện hàm checkLogin().

4. Lớp User trả về kết quả cho lớp LoginView.
5. Lớp LoginView gọi lớp DataEntryView.
6. Lớp DataEntryView hiển thị cho NVNL.
7. NVNL chọn chức năng Cập nhật kết quả từ lớp DataEntryView.
8. Lớp DataEntryView gọi lớp UpdateResultView.
9. Lớp UpdateResultView hiển thị cho NVNL.
10. NVNL click vào ComboBox Chặng đua.
11. Lớp UpdateResultView gọi lớp RaceTrack để xử lý.
12. Lớp RaceTrack thực hiện hàm getRaceTrackList().
13. Lớp RaceTrack trả về kết quả cho lớp UpdateResultView.
14. Lớp UpdateResultView hiển thị kết quả cho NVNL.
15. NVNL chọn một chặng đua ở lớp UpdateResultView.
16. Lớp UpdateResultView gọi lớp RaceResult để xử lý.
17. Lớp RaceResult thực hiện hàm getRaceResultList().
18. Lớp RaceResult trả kết quả về cho lớp UpdateResultView.
19. Lớp UpdateResultView hiển thị kết quả cho NVNL.
20. NVNL nhập kết quả thi đấu(thời gian hoàn thành, số vòng đua hoàn thành cho từng tay đua) của tất cả tay đua tại giao diện lớp UpdateResultView. NVNL click nút Lưu.
21. Lớp UpdateResultView gọi lớp RaceResult để xử lý.
22. Lớp RaceResult thực hiện hàm updateRaceResult().
23. Lớp RaceResult trả về kết quả cho lớp UpdateResultView.
24. Lớp UpdateResultView thông báo cập nhật kết quả thành công cho NVNL.

b. Biểu đồ tuần tự pha phân tích



6. Biểu đồ thiết kế lớp thực thể của modul

a. Diễn giải

* Bước 1: Hoàn thiện lớp và thuộc tính

- Lớp User:
 - id : int (Bổ sung)
 - username: String
 - password: String
 - fullName: String
 - role: String
- Lớp Driver:
 - id: int (Bổ sung)
 - name: String
 - dob: Date
 - national: String

- bio: String
- Lớp RaceTrack:
 - id: int (Bổ sung)
 - name: String
 - location: String
 - numOfLaps: int
 - time: Date
 - des: String
- Lớp RaceTeam:
 - id: int (Bổ sung)
 - name: String
 - brand: String
 - des: String
- Lớp Membership:
 - id: int (Bổ sung)
 - startTime: Date
 - endTime: Date
- Lớp RaceResult:
 - id: int (Bổ sung)
 - completionTime: Duration
 - completedLaps: int
 - score: int

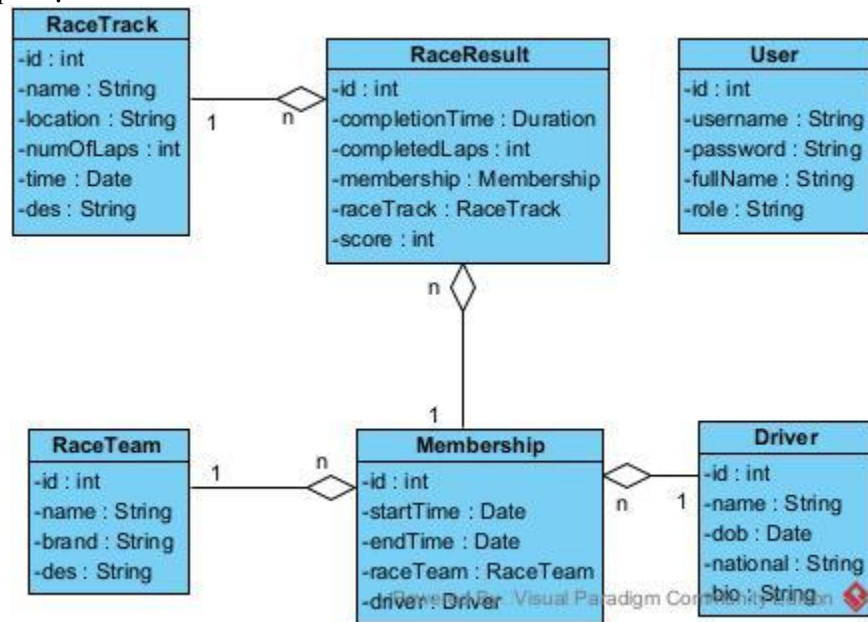
*** Bước 2:** Chuyển quan hệ Association

- Driver + RaceTeam → Membership được chuyển thành: thông tin Driver nằm trong Membership, thông tin RaceTeam nằm trong Membership.
- Membership + RaceTrack → RaceResult được chuyển thành: thông tin Membership nằm trong RaceResult, thông tin RaceTrack nằm trong RaceResult.

*** Bước 3:** Bổ sung thuộc tính đối tượng

- RaceTeam nằm trong Membership, quan hệ 1 – n → Membership có một RaceTeam
- Driver nằm trong Membership, quan hệ 1 – n → Membership có một Driver
- RaceTrack nằm trong RaceResult, quan hệ 1 – n → RaceResult có một RaceTrack
- Membership nằm trong RaceResult, quan hệ 1 – n → RaceResult có một Membership

b. Biểu đồ lớp thực thể



7. Biểu đồ thiết kế CSDL của modul

a. Diễn giải

* **Bước 1:** Mỗi lớp thực thể tạo một bảng tương ứng

- Driver → tblDriver
- RaceTeam → tblRaceTeam
- RaceTrack → tblRaceTrack
- User → tblUser
- Membership → tblMembership
- RaceResult → tblRaceResult

* **Bước 2:** Chuyển các thuộc tính không phải đối tượng của thực thể thành tên cột của bảng tương ứng.

- tblDriver: id, name, dob, national, bio,
- tblRaceTeam: id, name, brand, des.
- tblRaceTrack: id, name, location, numOfLaps, time, des.
- tblUser: id, username, password, fullName, role.
- tblMembership: id, startTime, endTime.
- tblRaceResult: id, completionTime, completedLaps, score.

* **Bước 3:** Chuyển quan hệ số lượng giữa các lớp thực thể thành quan hệ số lượng giữa các bảng.

- tblDriver – tblMembership: 1 – n
- tblRaceTeam – tblMembership: 1 – n
- tblRaceTrack – tblRaceResult: 1 – n

- tblMembership – tblRaceResult: 1 – n

*** Bước 4:** Bổ sung thuộc tính khóa

Khóa chính được thiết lập với thuộc tính id của các bảng tương ứng.

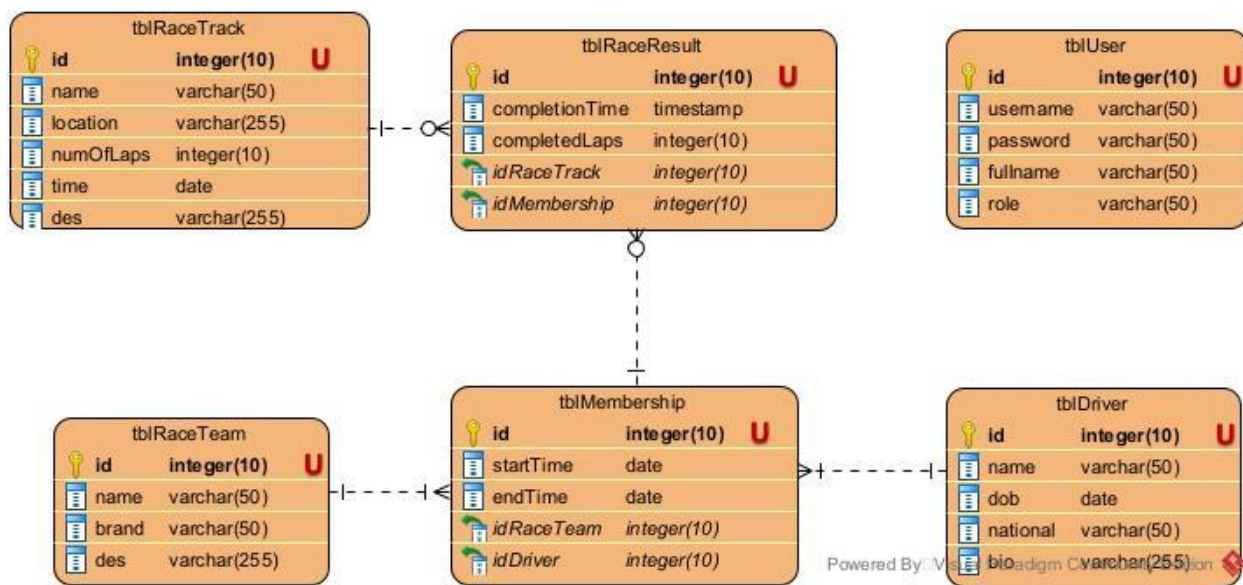
Khoái ngoại được thiết lập như sau:

- tblDriver – tblMembership: 1 – n → bảng tblMembership có khóa ngoại idDriver tham chiếu đến tblDriver.
- tblRaceTeam – tblMembership: 1 – n → bảng tblMembership có khóa ngoại idRaceTeam tham chiếu đến tblRaceTeam.
- tblRaceTrack – tblRaceResult: 1 – n → bảng tblRaceResult có khóa ngoại idRaceTrack tham chiếu đến tblRaceTrack.
- tblMembership – tblRaceResult: 1 – n → bảng tblRaceResult có khóa ngoại idMembership tham chiếu đến tblMembership.

*** Bước 5:** Loại bỏ thuộc tính dư thừa

- Trong tblRaceResult: thuộc tính score có thể được suy diễn ra từ completionTime và completedLaps → thuộc tính dẫn xuất → loại bỏ

b. Kết quả biểu đồ thiết kế CSDL



8. TK giao diện và biểu đồ lớp thiết kế chi tiết đầy đủ của modul

a. Thiết kế giao diện

*** Giao diện đăng nhập**

The image shows a simple login form. At the top, the word 'Login' is centered. Below it, there are two text input fields. The first is labeled 'username' and the second is labeled 'password'. Below the password field is a button labeled 'Login'.

Lớp biên: LoginFrm

Thuộc tính tường minh:

- txtUsername : JTextFeild
- txtPassword : JPasswordField
- btnLogin : Jbutton

Thuộc tính ẩn: Không có

Đề xuất các hàm xử lý:

1. Xử lý kiểm tra đăng nhập → Đề xuất một phương thức:
 - Name: checkLogin()
 - Input: username, password (User)
 - Ứng viên tham số vào (input):
 - checkLogin(username: String) → loại do không đủ thông tin đầu vào
 - checkLogin(password: String) → loại do không đủ thông tin đầu vào
 - checkLogin(List<User>) → loại do xử lý đầu vào cồng kềnh
 - checkLogin(username: String, password: String) → input là 2 thuộc tính của lớp User nên đề xuất tham số vào là User
 - checkLogin(user : User) → Chọn
 - Output:
 - Xác minh đăng nhập đúng hay không (boolean)
 - Xác định vị trí công việc (role) của nhân viên để gọi trang chủ tương ứng của nhân viên.
 - Ứng viên tham số ra (output):
 - checkLogin(user : User): boolean

→ nếu trả về true thuộc tính role, fullName, id được đưa vào tham số user đầu vào.

checkLogin(user : User): String

→ login thành công thì trả về String “true”, các thuộc tính role, fullName, id được đưa vào tham số user đầu vào. Ngược lại thì trả về “false”.

checkLogin(user : User): void

→ login thành công thì tham số user đầu vào trả về khác null, các thuộc tính role, fullName, id được đưa vào tham số user đầu vào. Ngược lại thì user trả về null.

checkLogin(user : User): int

→ login thành công thì trả về số > 0, các thuộc tính role, fullName, id được đưa vào tham số user đầu vào. Ngược lại thì trả về 0.

checkLogin(user : User): User

→ login thành công thì trả về đối tượng User khác null với dữ liệu đã được set, ngược lại thì trả về null. (loại vì truyền vào User nên không cần trả về User nữa)

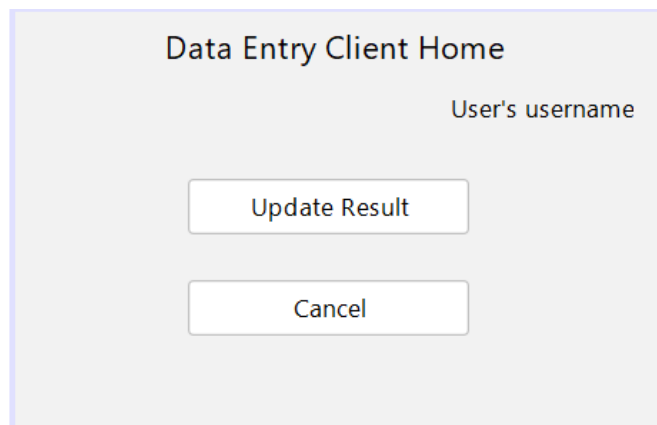
checkLogin(user : User): List<User>

→ login thành công thì trả về danh sách User với đầy đủ thuộc tính, ngược lại thì trả về null. (loại vì truyền vào User nên không cần trả về User nữa)

→ Chọn checkLogin(user : User): boolean vì là phương án tối ưu nhất, không phải giải thích phức tạp

- Gán lớp DAO: UserDao

*** Giao diện chính của Nhân viên nhập liệu:**



Lớp biên: DataEntryHomeFrm()

Thuộc tính tường minh:

- btnUpdateResult: JButton
- btnCancel: JButton

Thuộc tính ẩn:

- user: User

* Giao diện cập nhật kết quả

The screenshot shows a Java Swing window titled "Update Result". Inside the window, there is a label "Chặng đua" followed by a dropdown menu currently showing "Item 1". Below this is a table with 8 columns: "STT", "Tên", "Tên đội đua", "Quốc tịch", "Ngày sinh", "Tiểu sử", "Thời gian hoàn...", and "Số vòng hoàn ...". The table body is currently empty. At the bottom right of the window, there is a button labeled "Lưu".

Lớp biên: UpdateResultFrm()

Thuộc tính tường minh:

- cbxRaceTrack: JComboBox
- tblRaceResult: JTable
- btnSave: JButton

Thuộc tính ẩn:

- user: User

Đề xuất các hàm xử lý:

1. Xử lý lấy danh sách chặng đua→ Đề xuất một phương thức:

- Name: getRaceTrackList()
- Input: Không có
- Output: Danh sách chặng đua

Ứng viên output:

RaceTrack[]

List<RaceTrack>

ArrayList<RaceTrack>

Vector<RaceTrack>

→ Chọn ArrayList<RaceTrack>

- Gán cho lớp DAO: RaceTrackDAO

2. Xử lý lấy danh sách kết quả của các tay đua tham gia chặng đua → Đề xuất một phương thức:

- Name: getRaceResultList()

- Input: getRaceResultList (idRaceTrack: int)

- Output: Danh sách kết quả → Chọn ArrayList<RaceResult>

- Gán cho lớp DAO: RaceResultDAO

3. Xử lý cập nhật kết quả tay đua → Đề xuất một phương thức

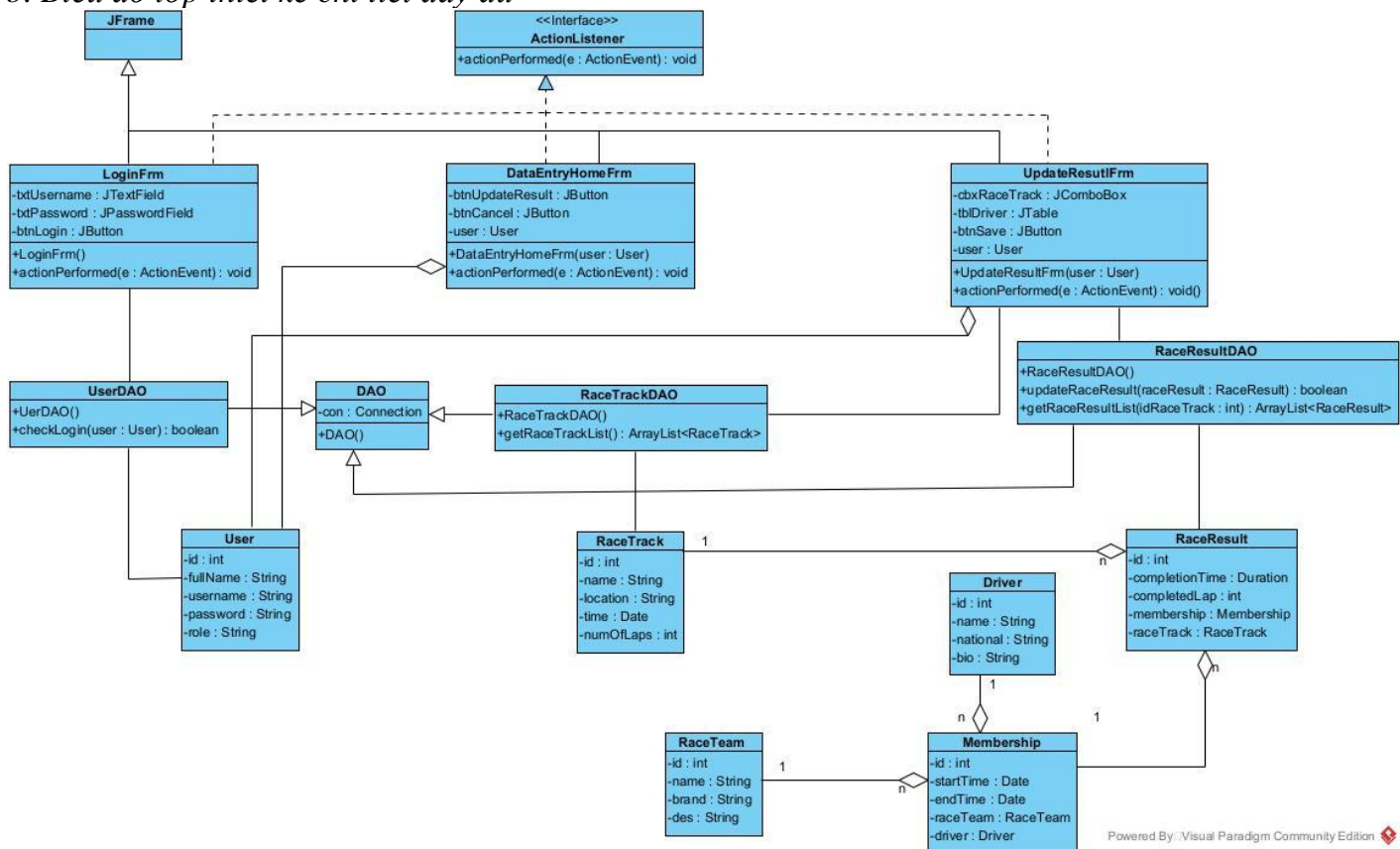
- Name: updateRaceResult()

- Input: updateRaceResult(raceResult: RaceResult)

- Output: boolean → Cập nhật thành công thì trả về true, ngược lại trả về false.

- Gán cho lớp DAO: RaceResultDAO

b. Biểu đồ lớp thiết kế chi tiết đầy đủ



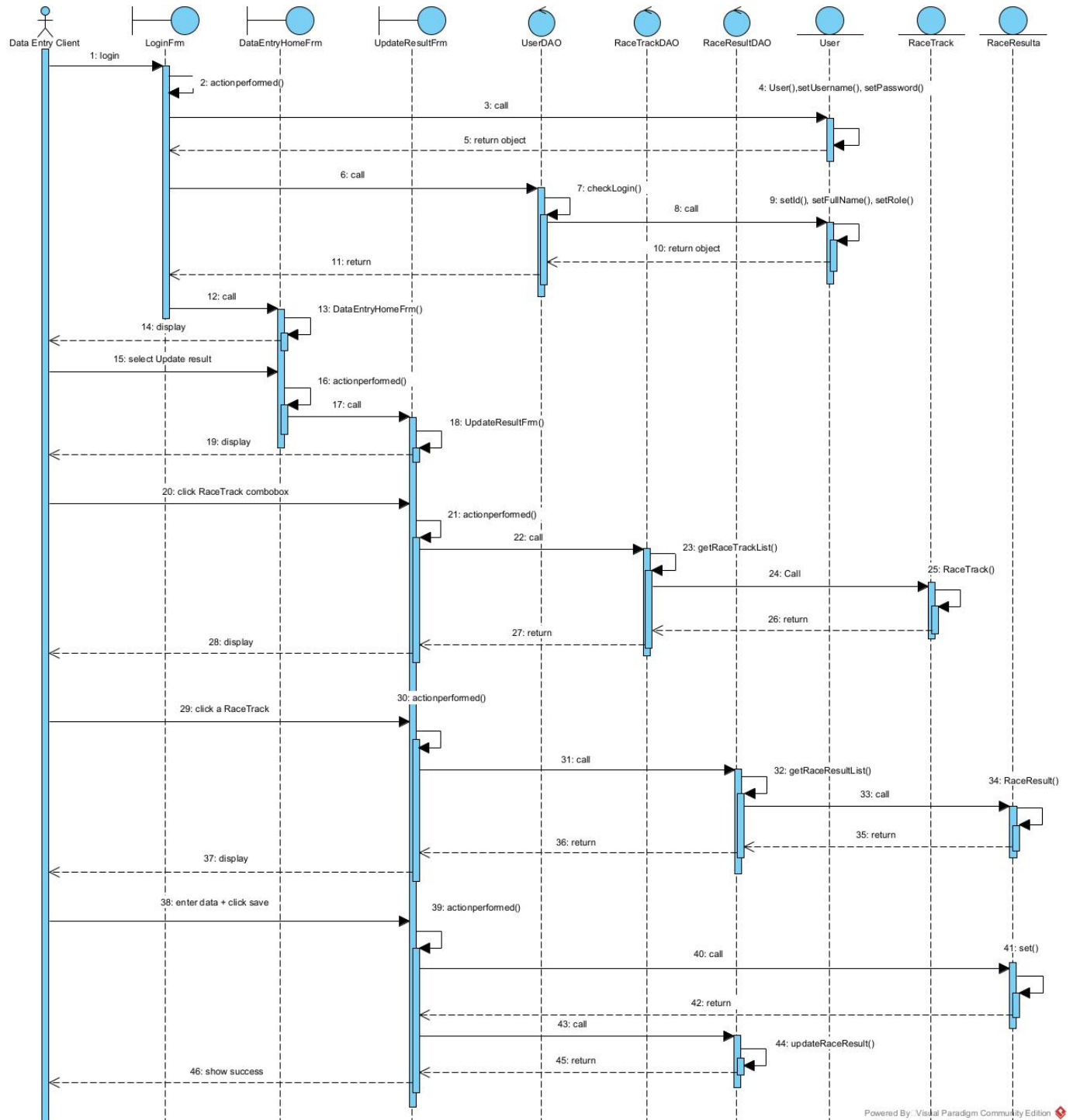
9. Biểu đồ tuần tự pha thiết kế của modul

a. Diễn giải kịch bản chuẩn v3

1. NVNL nhập username, password và ấn nút Login ở LoginFrm.
2. Hàm actionPerformed() của LoginFrm được gọi
3. Hàm actionPerformed() gọi User để tạo đối tượng User
4. Lớp User đóng gói thông tin vào đối tượng User
5. Lớp User trả về đối tượng User cho hàm actionPerformed()
6. Hàm actionPerformed() gọi hàm checkLogin() của lớp UserDao
7. Hàm checkLogin() kiểm tra thông tin đăng nhập.
8. Hàm checkLogin() gọi lớp User để gán thêm các thuộc tính id, fullName, role.
9. Lớp User gọi hàm setId(), setFullName(), setRole()
10. Lớp User trả về đối tượng UserR cho hàm checkLogin()
11. Hàm checkLogin() trả kết quả về cho hàm actionPerformed()
12. Hàm actionPerformed() gọi lớp DataEntryHomeFrm
13. Hàm khởi tạo DataEntryHomeFrm() được gọi
14. Giao diện DataEntryHomeFrm được hiển thị cho NVNL
15. NVNL click nút Update Result tại DataEntryHomeFrm
16. Hàm actionPerformed() của DataEntryHomeFrm được gọi
17. Hàm actionPerformed() gọi lớp UpdateResultFrm
18. Hàm khởi tạo UpdateResultFrm() được gọi
19. Giao diện UpdateResult được hiển thị cho NVNL
20. NVNL click chọn combobox chặng đua tại UpdateResultFrm
21. Hàm actionPerformed() của UpdateResultFrm được gọi
22. Hàm actionPerformed() gọi đến hàm getRaceTrackList() của lớp RaceTrackDAO
23. Hàm getRaceTrackList () lấy danh sách các chặng đua.
24. Hàm getRaceTrackList () gọi lớp RaceTrackDAO để đóng gói các kết quả tìm được.
25. Lớp RaceTrack đóng gói thông tin thành đối tượng RaceTrack
26. Lớp RaceTrack trả về đối tượng RaceTrack cho hàm getRaceTrackList ()
27. Hàm getRaceTrackList () trả về kết quả cho hàm actionPerformed()
28. Giao diện UpdateResultFrm hiển thị kết quả cho NVNL
29. NVNL click chọn một chặng đua
30. Hàm actionPerformed() của UpdateResultFrm được gọi
31. Hàm actionPerformed() gọi hàm getRaceResultList() của RaceResultDAO
32. Hàm getRaceResultList() lấy danh sách thông tin các kết quả của chặng đua
33. Hàm getRaceResultList() gọi lớp RaceResult để đóng gói thông tin
34. Lớp RaceResult đóng gói thông tin thành đối tượng RaceResult

35. Lớp RaceResult trả về đối tượng cho hàm getRaceResultList ()
36. Hàm getRaceResultList() trả về kết quả cho hàm actionPerformed()
37. Giao diện UpdateResultFrm hiển thị kết quả cho NVNL
38. NVNL nhập thông tin kết quả tay đua và click nút Save
39. Hàm actionPerformed() của UpdateResultFrm được gọi
40. Hàm actionPerformed() gọi lớp RaceResult để set các thuộc tính được sửa đổi.
41. Lớp RaceResult set các thuộc tính được sửa đổi vào đối tượng RaceResult
42. Lớp RaceResult trả về đối tượng cho hàm actionPerformed()
43. Hàm actionPerformed() gọi hàm updateRaceResult() của lớp RaceResultDAO
44. Hàm updateRaceResult() được thực thi.
45. Hàm updateRaceResult() trả về kết quả cho hàm actionPerformed()
46. Giao diện UpdateResultFrm hiển thị thông báo cho NVNL.

b. Biểu đồ tuần tự pha thiết kế



10. Test plan và test case chuẩn cho test hộp đen của modul

a. Blackbox test plan

STT	Tên module	Test case
1	Cập nhật kết quả	Cập nhật RaceResult đã tồn tại
2		Cập nhật RaceResult chưa tồn tại
3		Cập nhật 2 lần RaceResult đã tồn tại

b. Testcase Chuẩn

***CSDL trước khi test:**

tblUser

id	username	password	fullname	role
1	a	a	A	Data Entry Client

tblRaceTrack

id	name	location	numOfLaps	time	des
1	Danang Race	Quận Hải Châu, TP Đà Nẵng	57	2025-04-25 14:00:00	Null
2	Hanoi Grand Fix	Quận Long Biên, TP Hà Nội	57	2025-05-10 14:00:00	Null
3	Ho Chi Minh City Grand Prix	Quận 1, TP Hồ Chí Minh	57	2025-06-15 08:00:00	Null

TblRaceTeam

id	name	brand	des
1	Red Velocity	Ferrari	Null
2	Storm Racers	Mercedes	Null
3	Thunderbolt Racing	Honda	Null

tblDriver

id	name	dob	national	bio
1	Ngô Tuấn Anh	2004-10-22	Việt Nam	Null
2	Vũ Trọng Khôi	2004-10-21	Việt Nam	Null
3	Lê Hải Đăng	1999-05-19	Việt Nam	Null
4	Phạm Ngọc Long	1991-06-24	Việt Nam	Null
5	Mai Xuân Nhân	1995-02-05	Việt Nam	Null

6	Nguyễn Thế Lâm	1997-10-10	Việt Nam	Null
---	----------------	------------	----------	------

tblMembership

id	idRaceTeam	idDriver	startTime	endTime
1	1	4	2025-01-01	2025-04-30
2	1	1	2025-01-01	2026-01-01
3	1	2	2025-01-01	2026-01-01
4	2	3	2025-05-01	2026-01-01
5	2	4	2025-05-01	2026-01-01
6	3	5	2025-01-01	2026-01-01
7	3	6	2025-01-01	2025-04-30
8	2	6	2025-05-01	2026-01-01

TblRaceResult

id	idRaceTrack	idMembership	completionTime	completedLaps
1	1	6	1:34:56.942	57
2	1	7	1:35:56.942	57
3	1	2	1:35:57.942	57
4	1	3	1:35:55.942	57
5	2	2	Null	Null
6	2	3	Null	Null
7	2	4	Null	Null
8	2	8	Null	Null

*** Kịch bản**

Scenario	Expected Results
1. Login User: a Password: a Click Login	- Giao diện chính của Data Entry Client hiện ra: <ul style="list-style-type: none"> • Update Result • Cancel
2. Chọn Update Result	- GD chức năng cập nhật kết quả: <ul style="list-style-type: none"> • ComboBox Chặng đua <div>Chọn chặng đua</div>

- Bảng danh sách tay đua (chưa có dữ liệu) với Thời gian về đích và Số vòng hoàn thành là các ô nhập trống

STT	Tên	Tên đội đua	Quốc tịch	Ngày sinh	Tiểu sử	Thời gian về đích	Số vòng hoàn thành

- Nút Lưu

3. Click ComboBox chặng đua

- GD chức năng cập nhật kết quả:

- ComboBox chặng đua:

Chọn chặng đua
Hanoi Grand Fix
Danang Race

- Bảng danh sách tay đua

STT	Tên	Tên đội đua	Quốc tịch	Ngày sinh	Tiểu sử	Thời gian về đích	Số vòng hoàn thành

- Nút Lưu

4. Chọn chặng đua Hanoi Grand Fix

- GD Chức năng cập nhật kết quả:

- ComboBox chặng đua

Hanoi Grand Fix

- Bảng danh sách tay đua

STT	Tên	Tên đội đua	Quốc tịch	Ngày sinh	Tiểu sử	Thời gian về đích	Số vòng hoàn thành

	1	Ngô Tuấn Anh	Red Velocity	Việt Nam	2004-10-22			
	2	Vũ Trọng Khôi	Red Velocity	Việt Nam	2004-10-21			
	3	Lê Hải Đăng	Storm Racers	Việt Nam	1999-05-19			
	4	Nguyễn Thế Lâm	Storm Racers	Việt Nam	1997-10-10			
<ul style="list-style-type: none"> Nút Lưu 								
5. Nhập lần lượt Thời gian về đích - Số vòng hoàn thành theo thứ tự:		- Hiện thị thông báo cập nhật kết quả thành công						
Thời gian về đích	Số vòng hoàn thành							
1:35:54.934	57							
1:35:39.435	57							
1:35:55.708	57							
0	15							
Click nút Lưu								

*** CSDL thay đổi sau khi test**

TblRaceResult

id	idRaceTrack	idMembership	completionTime	completedLaps
1	1	6	1:34:56.942	57
2	1	7	1:35:56.942	57
3	1	2	1:35:57.942	57
4	1	3	1:35:55.942	57
5	2	2	1:35:54.934	57
6	2	3	1:35:39.435	57
7	2	4	1:35:55.708	57

8	2	8	0	15
---	---	---	---	----