ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGÀNH CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CUỐI KỲ

NHÓM

Phan Ngọc Thiên Ân 18127037

Hoàng Nguyên Trúc 18127055

Phạm Xuân Tú 18127056

Môn học: An toàn và bảo mật hệ thống thông tin

Thành phố Hồ Chí Minh – 2021

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGÀNH CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CUỐI KỲ

GIÁO VIÊN HƯỚNG DẪN

ThS. Lương Vĩ Minh

TS. Phạm Thị Bạch Huệ

Môn học: An toàn và bảo mật hệ thống thông tin

Thành phố Hồ Chí Minh – 2021

# LỜI CẢM ƠN

Đầu tiên, chúng em xin cảm ơn Trường Đại học Khoa Học Tự Nhiên Thành phố Hồ Chí Minh đã tạo điều kiện giúp chúng em có thể hoàn thành đồ án này.

Chúng em cũng xin cảm ơn sự trợ giúp của 2 thầy cô phụ trách là ThS. Lương Vĩ Minh và TS. Phạm Thị Bạch Huệ đã nhiệt tình khuyến khích và trợ giúp chúng em để chúng em có thể hoàn thành đồ án một cách thuận lợi.

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc72828007)

[MỤC LỤC 3](#_Toc72828008)

[BẢNG PHÂN CÔNG CÔNG VIỆC 4](#_Toc72828009)

[PHÂN HỆ 2 6](#_Toc72828010)

[2.1) Những chính sách bảo mật đã học: 6](#_Toc72828011)

[2.1.1) User authentication: chính sách định danh người dùng. 6](#_Toc72828012)

[2.1.2) Access Control: 7](#_Toc72828013)

[2.1.3) Encryption: phương pháp mã hóa dữ liệu. 8](#_Toc72828014)

[2.1.4) Audit: cơ chế ghi nhật ký trong cơ sở dữ liệu. 9](#_Toc72828015)

[2.2) Những chính sách bảo mật nhóm đã cài đặt: 9](#_Toc72828016)

[2.2.1) CBAC: 9](#_Toc72828017)

[2.2.2) RBAC: 10](#_Toc72828018)

[2.2.3) VPD: 11](#_Toc72828019)

[2.2.4) Encryption: 12](#_Toc72828020)

[2.2.5) OLS: 14](#_Toc72828021)

[2.2.6) Audit: 14](#_Toc72828022)

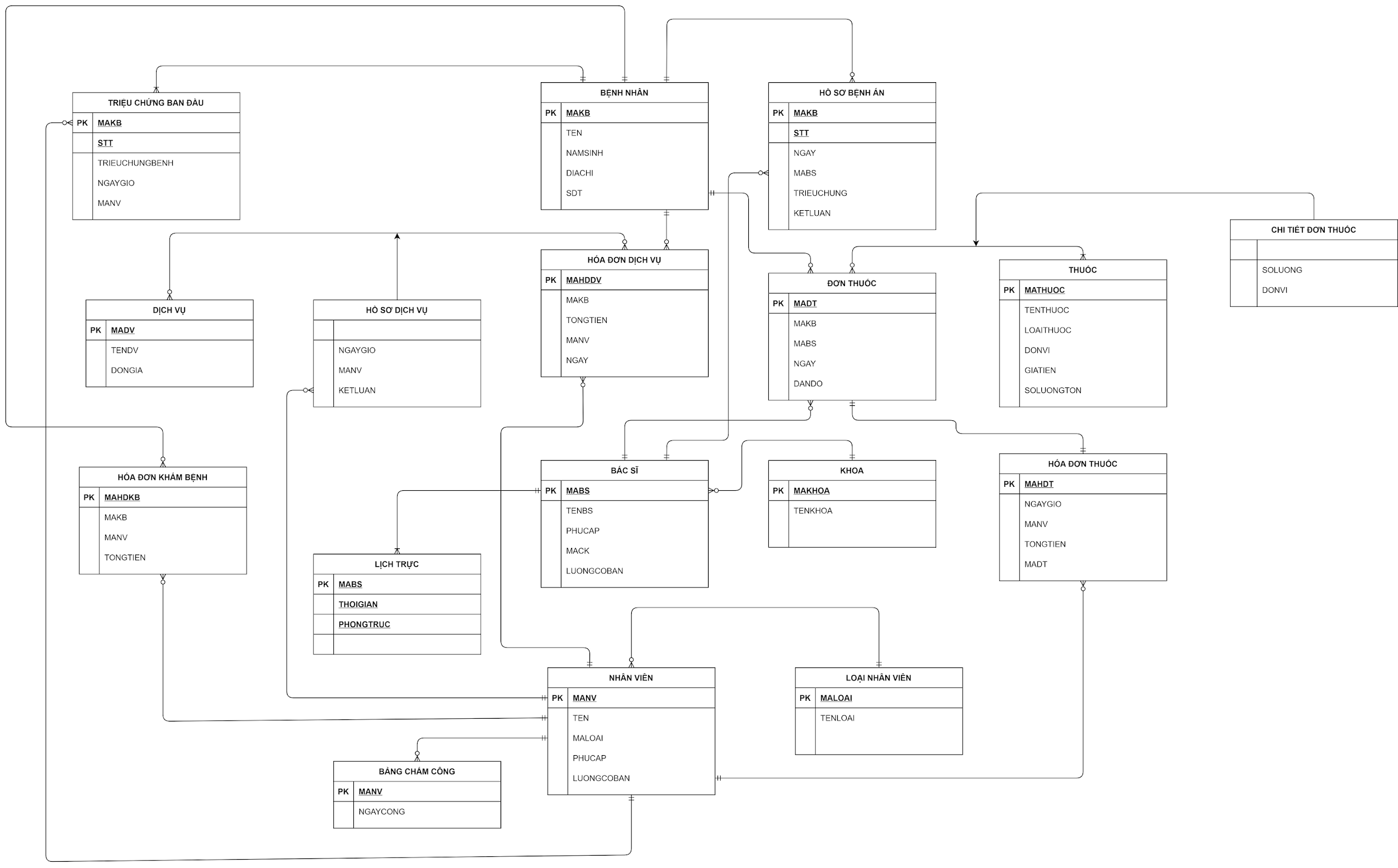
[TÀI LIỆU THAM KHẢO 16](#_Toc72828023)

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| MSSV | TÊN | CÔNG VIỆC | ĐỘ HOÀN THÀNH |
| 18127055 | Hoàng Nguyên Trúc | Cài đặt UI | 100 |
| Xem danh sách người dùng trong hệ thống | 100 |
| Thông tin về quyền (privileges) của mỗi user/ role trên các đối tượng dữ liệu | 100 |
| Cho phép tạo mới, xóa, sửa (hiệu chỉnh) user | 100 |
| Cho phép thực hiện việc cấp quyền: cấp quyền cho user, cấp quyền cho role, cấp role cho user. Quá trình cấp quyền có tùy chọn là có cho phép người được cấp quyền có thể cấp quyền đó cho user/ role khác hay không (có chỉ định WITH GRANT OPTION hay không) | 100 |
| Cho phép thu hồi quyền từ người dùng/ role | 100 |
| Xem danh sách các đối tượng hiện có trên cơ sở dữ liệu (table, user, role, view) | 100 |
| Viết báo cáo | 100 |
| Hàm đăng nhập/đăng xuất | 100 |
| Cài đặt chính sách mã hóa | 100 |
| Xem danh sách các đối tượng hiện có trong hệ thống (view, table, role) | 100 |
| Cài đặt chính sách RBAC (role Bác sĩ) | 100 |
| Phân tích OLS | 100 |
| 18127056 | Phạm Xuân Tú | Lập trình script tạo bảng | 100 |
| Cho phép kiểm tra quyền của các chủ thể vừa được cấp quyền | 100 |
| Cài đặt chính sách RBAC (role Tiếp tân) | 100 |
| Cài đặt chính sách VPD – Kịch bản 2 | 100 |
| 18127037 | Phan Ngọc Thiên Ân | Quyền select, update thì cho phép phân quyền tính đến mức cột | 100 |
| Cho phép tạo/xóa role | 100 |
| Viết báo cáo | 90 |
| Lược đồ cơ sở dữ liệu | 100 |
| Lập trình script tạo khóa giữa các bảng và tạo data | 100 |
| Cài đặt chính sách VPD – Kịch bản 1 | 100 |
| Cài đặt audit cơ bản và FGA | 100 |
| Cài đặt enable và disable audit | 100 |

# PHÂN HỆ 2

1. **Lược đồ cơ sở dữ liệu:**



1. **Chính sách bảo mật:**

## 2.1) Những chính sách bảo mật đã học:

2.1.1) User authentication: chính sách định danh người dùng.

* Được sử dụng với các hệ thống đóng.
* Thường được kết hợp thêm với các chính sách bảo mật khác để tăng tính an toàn.
* Các phương pháp định danh phổ biến:
* Username và password: password thường được lưu ở dạng hash (kết hợp định danh và mã hóa).
* Sinh trắc học (vân tay, mắt, ...)
* RFID: thẻ từ chứa thông tin.
* Token: 1 dạng chìa khóa, thẻ nhận dạng đã được cấp phép đăng nhập vào hệ thống.
* Mã OTP: loại mã được tạo ra và gửi cho người dùng khi có yêu cầu đăng nhập được gửi tới hệ thống, có giá trị trong 1 khoảng thời gian nhất định hoặc có giới hạn số lần sử dụng (thường là 1).
* Cảnh báo truy cập: hệ thống sẽ thông báo cho chủ tài khoản biết rằng có người đã đăng nhập vào tài khoản của mình và đợi sự xác nhận từ chủ tài khoản.
* Ưu điểm:
* Dễ dàng cài đặt và sử dụng.
* Cơ chế đơn giản nhưng hiệu quả, được sử dụng rộng rãi trên nhiều hệ thống vì độ đa dạng của phương pháp.
* Khuyết điểm:
* Giá thành không ổn định, có những phương pháp đắt tiền vì cần thêm các trang thiết bị chuyên dụng.
* Không thể ngăn người dùng lạm quyền một khi người dùng đã vào được hệ thống.

2.1.2) Access Control: chính sách điều khiển truy cập.

* DAC (Direct access control): là chính sách cơ bản nhất của việc phân quyền cho người dùng dựa vào tài khoản của chính người dùng đó, có thể phân quyền cho 1 người nhiều lần nhưng chỉ có thể phân quyền cho 1 người trong 1 lúc, có thể trao quyền hạn của mình cho người dùng khác. Thường sẽ được tích hợp chung với các chính sách điều khiển truy cập khác.
* CBAC (Content-based access control): là chính sách có cơ sở là việc tạo và phân chia các khung nhìn (view) cũng như các quyền hạn truy cập và sửa đổi các thuộc tính dữ liệu cho từng tài khoản người dùng .
* RBAC (Role-based access control): là chính sách lấy cơ sở gom nhóm nhiều tài khoản người dùng lại tạo thành 1 nhóm người dùng có cùng quyền hạn đọc, ghi đối với các đối tượng trong cơ sở dữ liệu, các tài khoản người dùng có cùng tính chất, chức vụ trong hệ thống sẽ được trao các quyền hạn giống nhau và có thể trao quyền hạn cho tất cả người dùng có chức vụ đó trong cùng 1 lúc.
* VPD (Virtual private database): một chính sách điều khiển truy cập đặc biệt của riêng Oracle, bằng cách gán các vị từ vào các câu truy vấn bảng khi được gọi dựa vào từng loại người dùng thông qua các policy được cài đặt trước, phương pháp này có thể giúp giảm thiểu số lượng view phải tạo nếu như đứng trước việc quá nhiều loại người dùng.
* OLS (Oracle level security): là hiện thực hóa của mô hình MAC (Mandatory access control), hoạt động dựa theo nguyên lý “no read up, no write down”. Phân quyền không dựa trên loại người dùng mà phân quyền dựa trên các cấp độ của dữ liệu. Được chia ra là 3 loại: Level (độ quan trọng của dữ liệu, cấp bậc chức vụ của người dùng), Compartment (các nhóm chức vụ bao gồm các người dùng có công việc giống nhau) và Group (thường sẽ là vị trí địa lý, các chi nhánh của tổ chức).
* Ưu điểm:
* Thích hợp cho các hệ thống lớn, có lượng người dùng và loại người dùng đông đảo.
* Giá thành cài đặt không cao.
* Tránh được việc người dùng vượt quyền, làm những việc không liên quan đến phận sự của mình.
* Khuyết điểm:
* Một người có thể nắm quá nhiều quyền hạn trong hệ thống (admin).
* Có thể xảy ra chuyện xung đột khi áp dụng nhiều chính sách điều khiển truy cập lên 1 bảng.
* Khó kiểm soát được việc người dùng trao quyền.

#### 2.1.3) Encryption: phương pháp mã hóa dữ liệu.

* Là phương pháp che dấu dữ liệu, biến dữ liệu từ có nghĩa sang vô nghĩa đối với các kẻ tấn công.
* Là rào cản cuối cùng của việc bảo mật dữ liệu, khi mà các chính sách bảo mật khác đều đã bị vượt qua.
* Lưu ý rằng phương pháp mã hóa chỉ có thể tránh được cuộc tấn công của người chứ không cản được cuộc tấn công của máy tính.
* 1 thuật toán mã hóa được gọi là an toàn khi: thời gian giải mã rất dài.
* Các phương pháp mã hóa phổ biến:
* Mã hóa đối xứng: mã hóa và giải mã đều xài chung 1 khóa.
* Mã hóa bất đối xứng: mã hóa bằng một public key nhưng để giải mã thì cần sử dụng 1 private key.
* Mã hóa lai: dùng phương pháp mã hóa bất đối xứng để mã hóa key dùng chung của mã hóa đối xứng.
* Hàm băm mật mã: dùng 1 key ban đầu để giấu dữ liệu.
* Ưu điểm:
* Là chính sách có thể nói là an toàn nhất vì có thể giấu được dữ liệu với cả DBA.
* Thông tin cho dù bị đọc cũng không bị lộ.
* Khuyết điểm:
* Mã hóa làm tăng lượng xử lý khi truy cập dữ liệu, tăng dung lượng lưu trữ dữ liệu.
* Làm hệ quản trị cơ sở dữ liệu không thể áp dụng các phương thức tìm kiếm dữ liệu cơ bản (index,...).
* Cần một chính sách quản lý và phân phối khóa thích hợp.
* Tất cả đều dựa vào khóa, vì thuật toán mã hóa được công bố rộng rãi nên nếu mã khóa bị lộ thì có nghĩa là dữ liệu sẽ bị lộ, nếu mất khóa thì xem như mất dữ liệu.

#### 2.1.4) Audit: cơ chế ghi nhật ký trong cơ sở dữ liệu.

* Được quản lý bởi admin hệ thống để quan sát và ghi lại các hoạt động cần thiết trong CSDL, tăng tính bảo mật và phục hồi dữ liệu.
* Thông tin được ghi gồm các thành phần: người thực hiện thao tác + thao tác dữ liệu + đối tượng dữ liệu + tình trạng của thao tác.
* Ưu điểm:
* Hỗ trợ admin hệ thống quản lý hệ thống một cách hiệu quả hơn, giúp quan sát rõ được từng người dùng đang làm gì nếu cần.
* Giúp ghi lại các hành động đáng ngờ trong hệ thống.
* Khi hệ thống bị lỗi hoặc mất mát dữ liệu thì nhật ký chính là cơ sở giúp hệ thống khôi phục lại các dữ liệu bị mất.
* Khuyết điểm:
* Bị sự chi phối của admin.
* Nếu audit lại tất cả hoạt động của hệ thống một cách không chọn lọc thì khó có thể biết được các thông tin đáng ngờ cũng như làm chậm hiệu suất hệ thống vì tăng thêm 1 việc là ghi nhật ký.

## 2.2) Những chính sách bảo mật nhóm đã cài đặt:

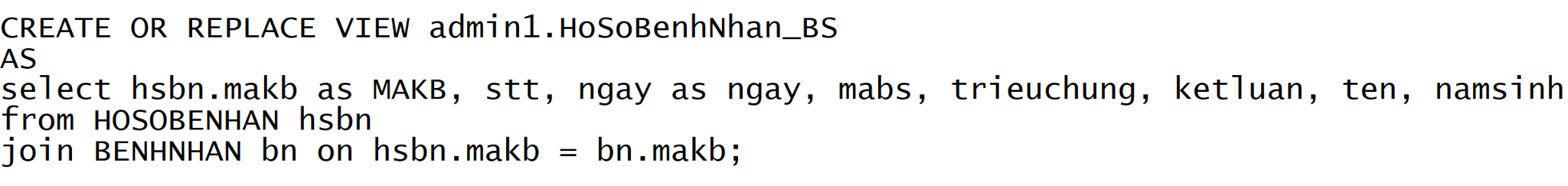
#### 2.2.1) CBAC:

* Kịch bản: Tạo view để những người dùng thuộc role “Bác sĩ” xem hồ sơ bệnh nhân .

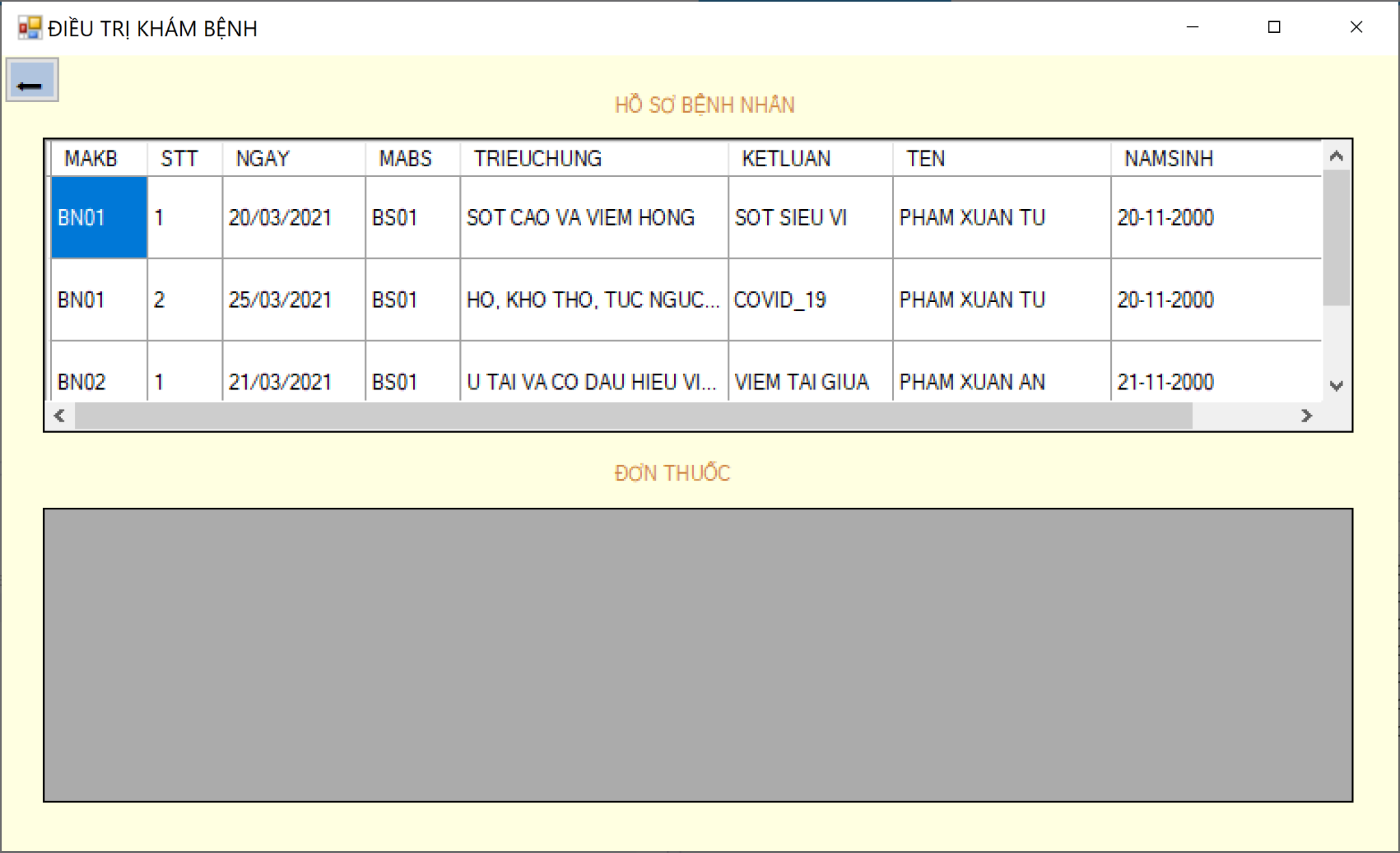
- Lý do sử dụng: Thay vì cấp quyền select trên 2 bảng hồ sơ bệnh nhân, bệnh nhân cho role “Bác sĩ” thì nhóm tạo view select kết 2 bảng sau đó cấp quyền select trên view vừa được tạo để vẫn có thể cung cấp khung nhìn thích hợp cho các bác sĩ, cũng vừa ngăn ngừa được việc xem thông tin trái quy định.

- Nhận xét, đánh giá, thuyết minh kết quả đạt được:

* Cài đặt view:



* Kết quả đạt được:



* Nhận xét: Sử dụng chính sách CBAC đáp ứng được yêu cầu đề ra.

#### 2.2.2) RBAC:

* Kịch bản 1: Những user được cấp role “Bác sĩ” được cấp quyền xem thông tin của bảng hồ sơ bệnh nhân.

- Lý do sử dụng: Vì số lượng nhân viên là bác sĩ trong một bệnh viện là rất nhiều nên việc gán quyền cho từng user sẽ mất thời gian. Chính vì thế nên việc tạo sẵn 1 role “Bác sĩ” và gán những quyền của role đó cho các user có vai trò bác sĩ sẽ tiết kiệm thời gian và thuận tiện cho việc cập nhật sau này hơn.

- Nhận xét: Sử dụng chính sách RBAC đáp ứng được yêu cầu đề ra. Những user được cấp role “Bác sĩ” đều có thể sử dụng những quyền mà role “Bác sĩ” được cấp.

* Kịch bản 2: Những user được cấp role “Tiếp tân và điều phối bệnh” được cấp quyền xem thông tin của bảng danh sách dịch vụ.

- Lý do sử dụng: Vì số lượng nhân viên là tiếp tân và điều phối bệnh trong một bệnh viện là một con số lớn nên việc gán quyền cho từng user sẽ tốn thời gian. Chính vì thế nên việc tạo sẵn 1 role “Tiếp tân và điều phối bệnh” và gán những quyền của role đó cho các user thuộc tiếp tân và điều phối bệnh thì sẽ nhanh hơn và tiện cho việc cập nhật sau này hơn.

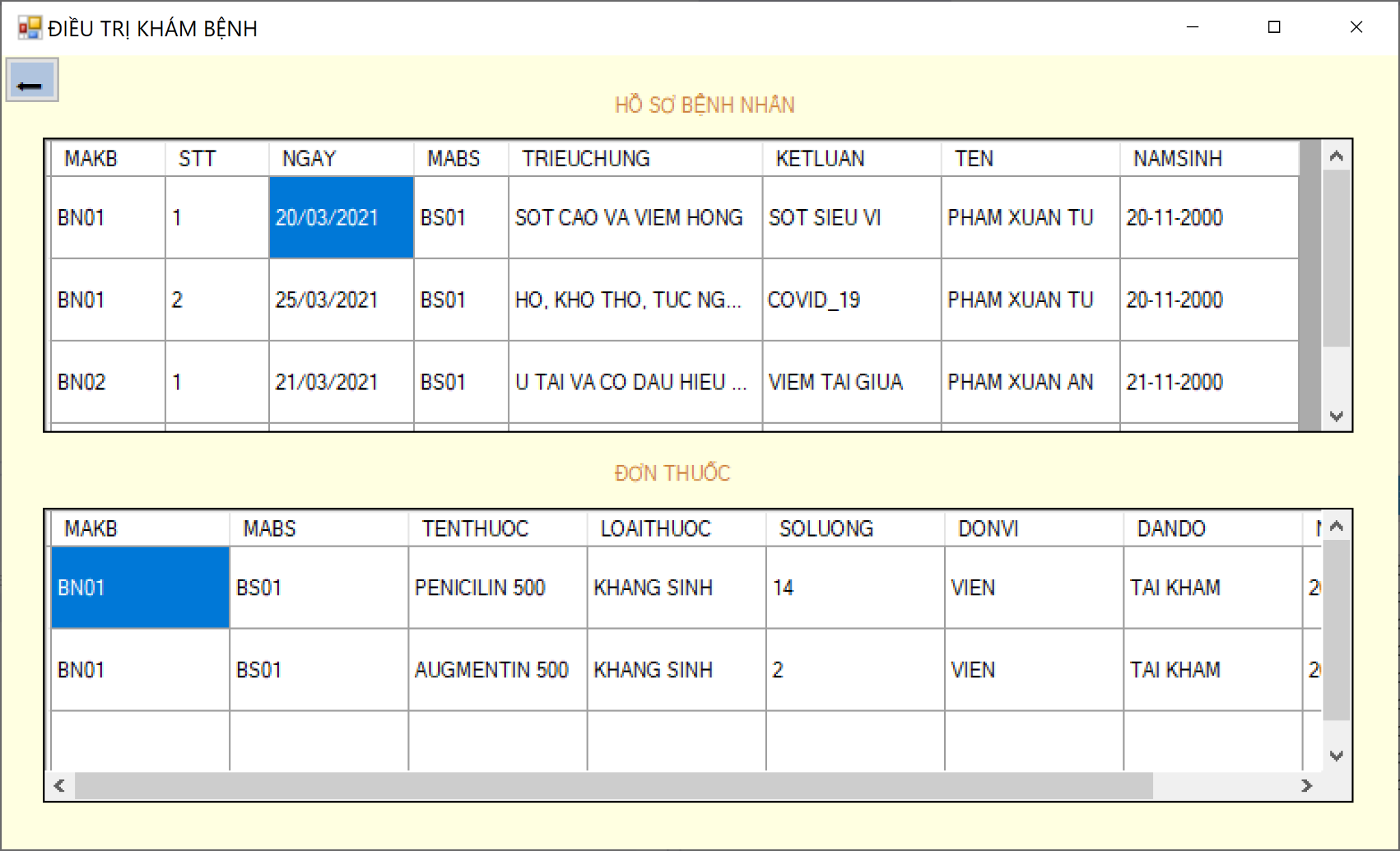
- Nhận xét: Sử dụng chính sách RBAC đáp ứng được yêu cầu đề ra. Những user được cấp role “Tiếp tân và điều phối bệnh” đều có thể sử dụng những quyền mà role “Tiếp tân và điều phối bệnh” được cấp.

#### 2.2.3) VPD:

* Kịch bản 1: Bác sĩ chỉ có thể xem thông tin liên quan đến việc điều trị bệnh và các loại thuốc phải dùng, liều dùng cho bệnh nhân mà bác sĩ chịu trách nhiệm điều trị. Bác sĩ không được xem thông tin khác của những bệnh nhân do bác sĩ khác điều trị.

- Lý do sử dụng: Kiểm soát cũng như ngăn chặn việc bác sĩ khác xem thông tin của bệnh nhân không phải do mình phụ trách, đó là điều trái pháp luật vì vi phạm luật bảo mật thông tin cá nhân trong bệnh viện.

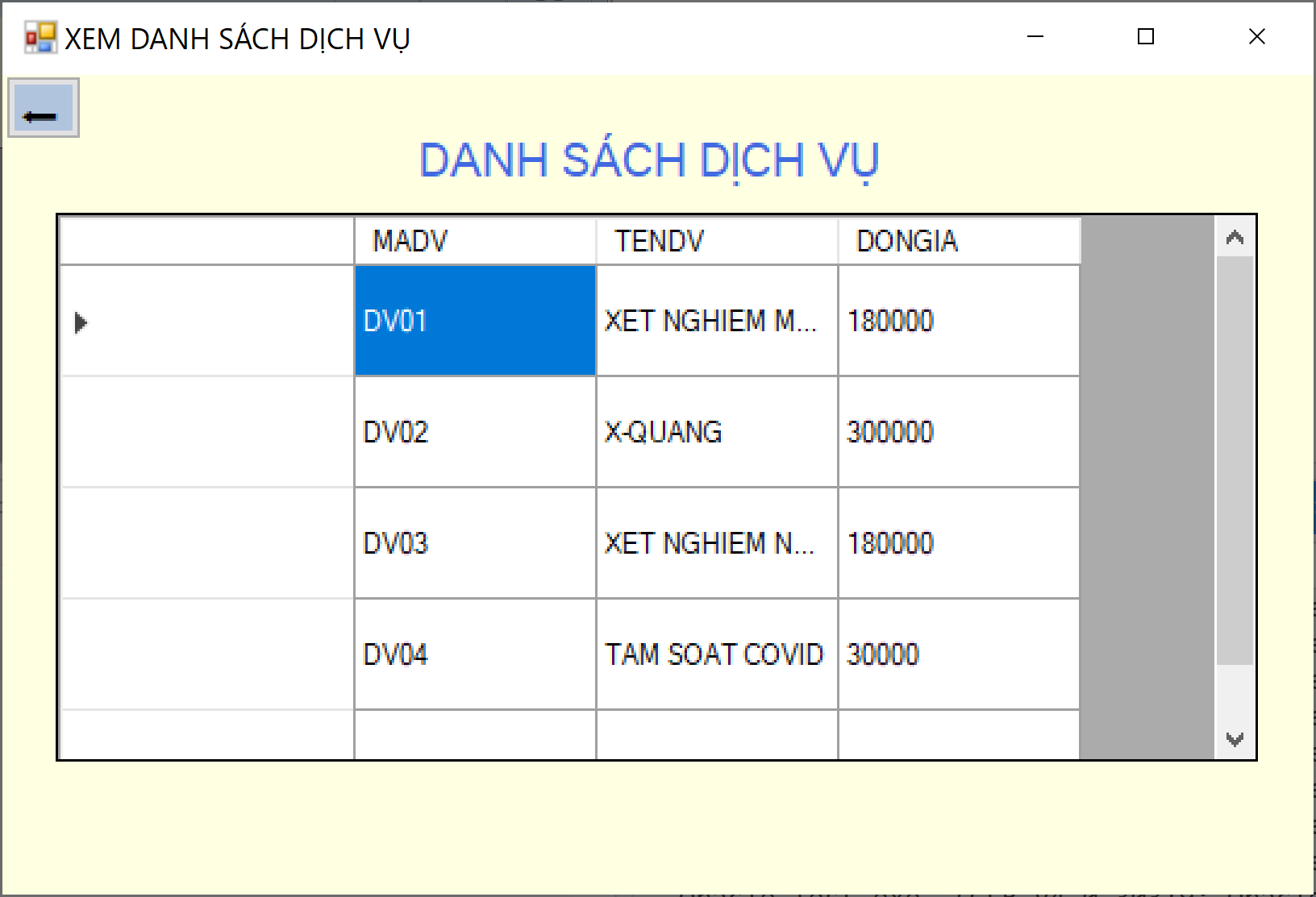
- Thuyết minh kết quả đạt được:



* Kịch bản 2: Bộ phận tiếp tân và điều phối bệnh có thể xem những dịch vụ hiện có nhưng không thể xem các thông tin liên quan đến số tiền cho từng dịch vụ.

- Lý do sử dụng: để có thể giấu được cột giá tiền thì ta có thể dùng view nhưng số lượng loại nhân viên lại nhiều nên việc tạo nhiều view và cấp sẽ phức tạp hơn việc ta gán thẳng tiêu chí lên trên bảng dịch vụ sẽ tiện lợi hơn nhiều.

- Thuyết minh kết quả đạt được:



#### 2.2.4) Encryption:

* Kịch bản: Thông tin của hồ sơ bệnh nhân là riêng tư, vì vậy hệ thống cần phải bảo mật thông tin này. Chính sách mã hóa sẽ được áp dụng cho trường dữ liệu “kết luận” trong bảng “hồ sơ bệnh nhân”.

- Lý do sử dụng:

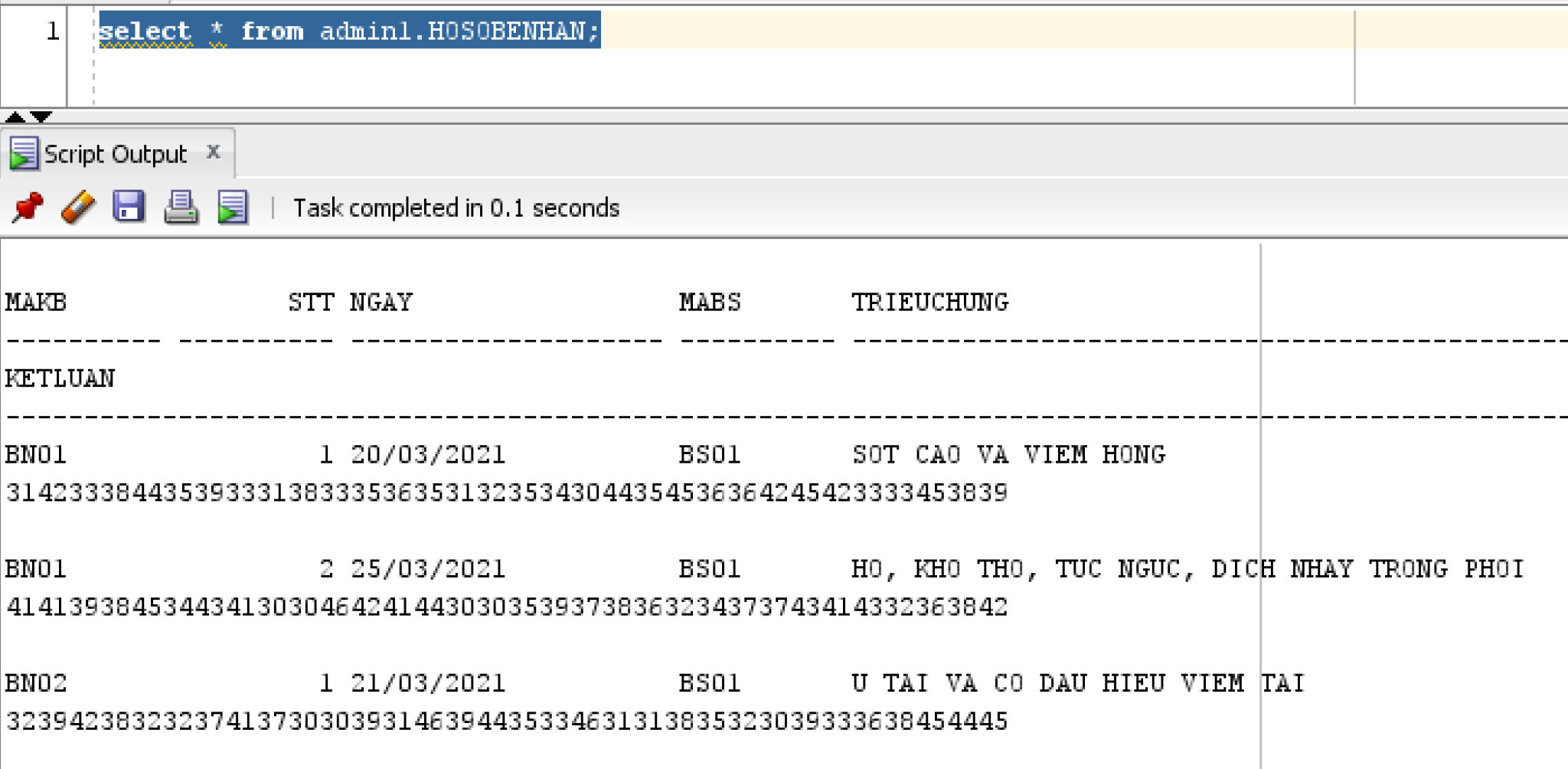
* Tuy đã áp dụng chính sách VPD để giới hạn chỉ cho phép những người dùng có vai trò là “bác sĩ” mới được phép truy cập tới dữ liệu của hồ sơ bệnh nhân mà họ chữa trị, nhưng để đề phòng trường hợp dữ liệu bị rò rỉ thì người đánh cắp cũng thể hiểu dữ liệu, chính sách mã hóa sẽ được áp dụng lên trường dữ liệu “kết luận” trong bảng “hồ sơ bệnh nhân”.

- Cách cài đặt:

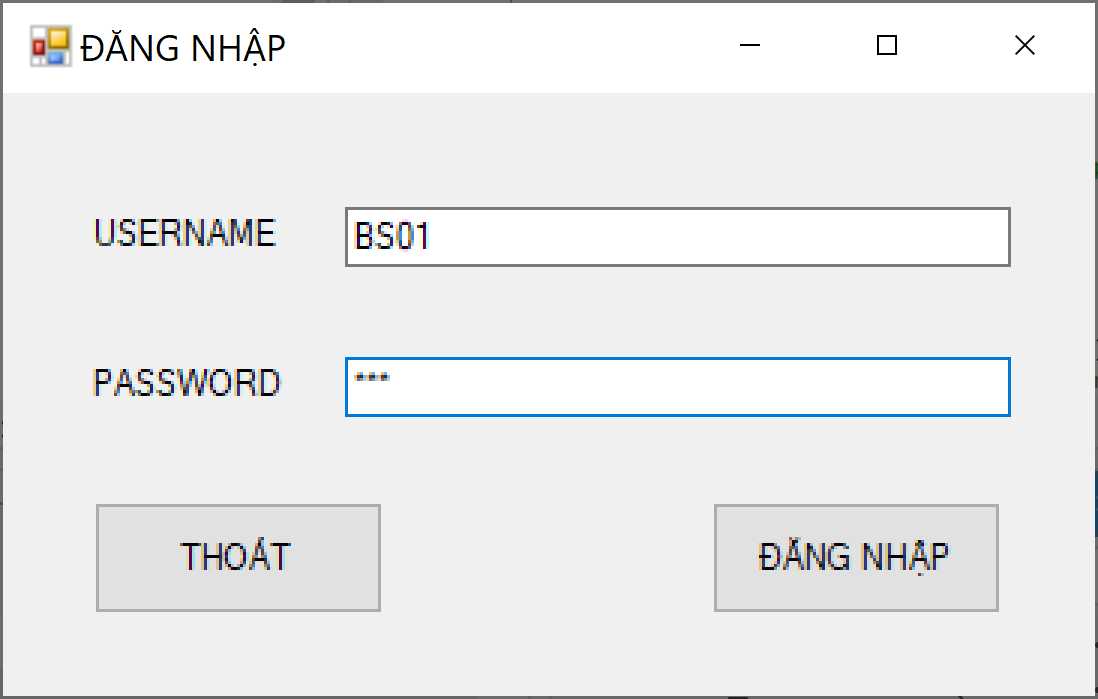
* Phương thức mã hóa: Mã hóa đối xứng. Khi dữ liệu được insert vào bảng “hồ sơ bệnh nhân”, dữ liệu của trường “kết luận” sẽ được mã hóa bằng 1 khóa bí mật, sau đó dữ liệu được lưu vào bảng “hồ sơ bệnh nhân”. Khi người dùng có vai trò là “bác sĩ” truy vấn đến bảng “hồ sơ bệnh nhân”, người dùng đó sẽ được tự động cấp khóa, dữ liệu sẽ được giải mã và hiển thị lên màn hình.
* Phương thức quản lý khóa: Khóa sẽ được lưu trong 1 bảng không có quan hệ với những bảng còn lại trong cơ sở dữ liệu.

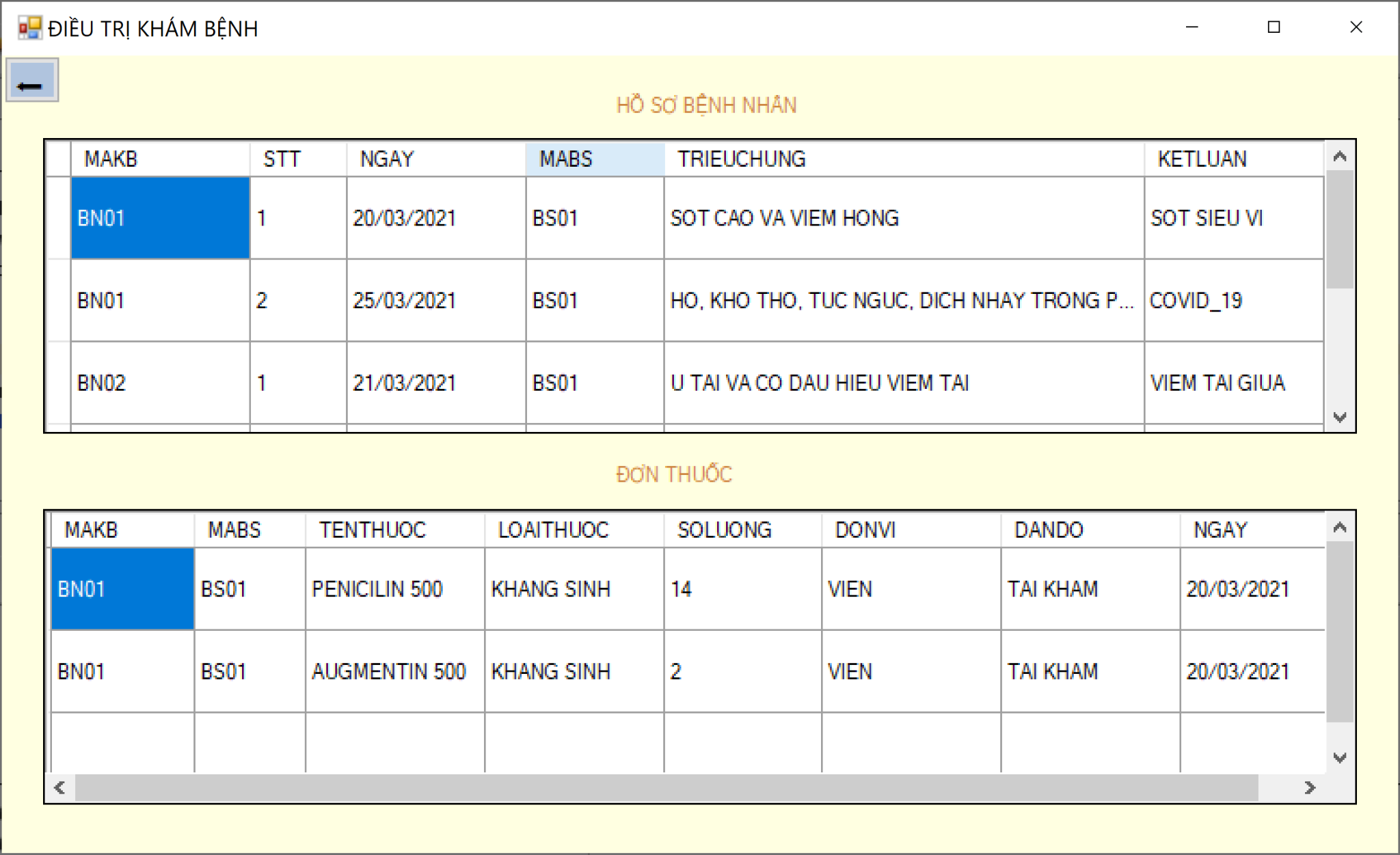
- Thuyết minh kết quả đạt được:

* Dùng tài khoản admin1 có vai trò “dba” để thử nghiệm:
* Khi người truy vấn đến bảng “hồ sơ bệnh nhân” không thuộc vai trò “bác sĩ”, dữ liệu sẽ hiển thị dưới dạng mã hóa:



* Dùng tài khoản BS01 có vai trò “bác sĩ” để thử nghiệm:
* Khi người truy vấn đến bảng “hồ sơ bệnh nhân” thuộc vai trò “bác sĩ”, dữ liệu liên quan đến bác sĩ đó sẽ được hiện đầy đủ:





#### 2.2.5) OLS:

* Kịch bản: Sử dụng OLS để gán label HIGHLY\_SENSIVTIVE:BACSI cho các user thuộc vai trò "Bác sĩ". Đối với user thuộc vai trò “Tiếp tân điều phối” sẽ có label SENSITIVE:DIEUPHOI, user thuộc vai trò “Admin” sẽ có label HIGHLY\_SENSITIVE: ADMIN. Dòng dữ liệu chứa khóa trong bảng “LOAIDICHVU” có OLS Label là: HIGHLY\_SENSIVTIVE (khóa dùng để mã hóa thông tin riêng tư của bệnh nhân mà chỉ cho phép bác sĩ phụ trách hồ sơ khám bệnh đó hoặc user có vai trò “Admin” được xem).

- Lý do sử dụng: Việc sử dụng OLS sẽ thêm vào 1 lớp bảo mật trong việc tránh các hành động đọc hay viết vượt quyền hạn của người dùng, qua đó có thể dễ dàng kiểm soát các hành động của các user thông qua các mức độ cũng như tính quan trọng của thông tin.

- Nhận xét, đánh giá, thuyết minh kết quả đạt được: Vì lý do kỹ thuật nên nhóm không cài đặt thành công OLS vào hệ thống.

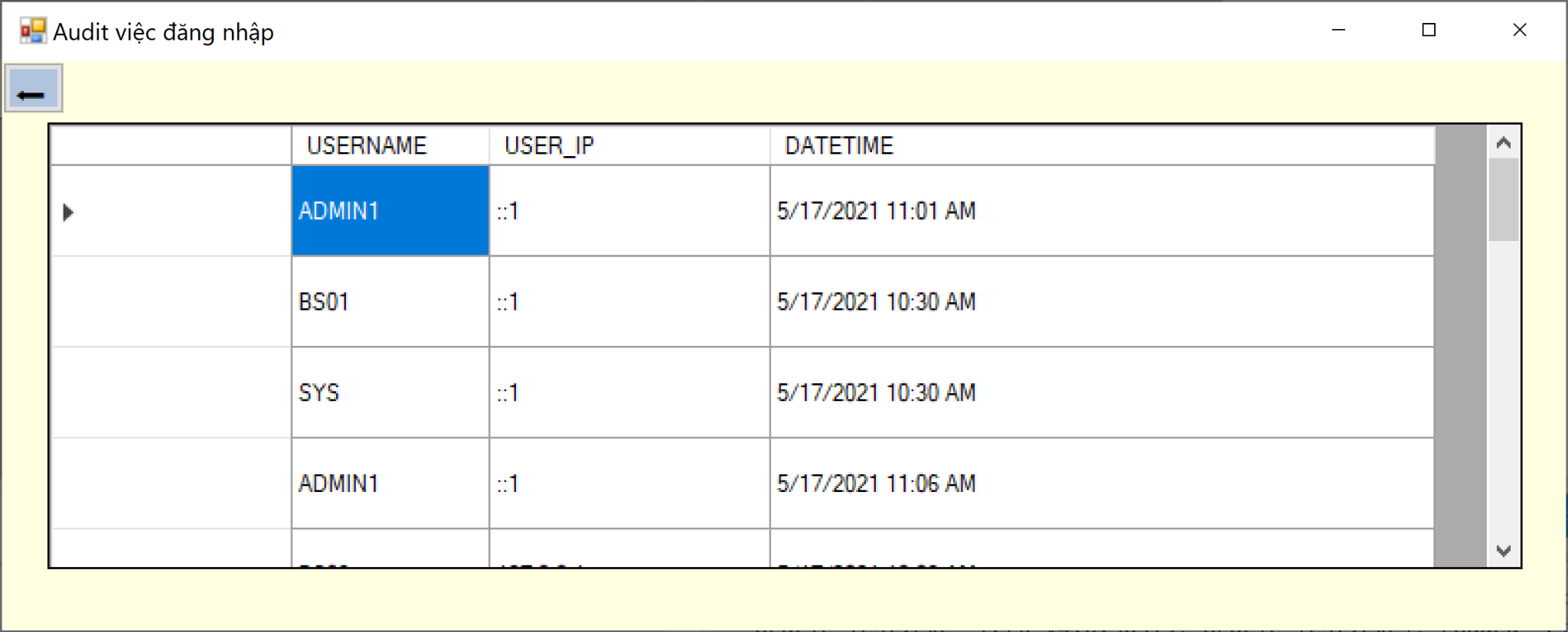
#### 2.2.6) Audit:

* Kịch bản 1 (Audit cơ bản): Hệ thống cần biết rõ thông tin đăng nhập của từng user trong hệ thống để có thể phát hiện các hành vi đăng nhâp trái phép từ những máy bên ngoài hệ thống.
* Kịch bản 2 (FGA): Các thông tin hồ sơ bệnh án của bệnh nhân cũng như thông tin đơn thuốc bệnh nhân sử dụng là các thông tin nhạy cảm, được bảo vệ theo pháp luật của nước Việt Nam, nên chuyện cho phép những người không phận sự xem các thông tin này là điều cấm kỵ.

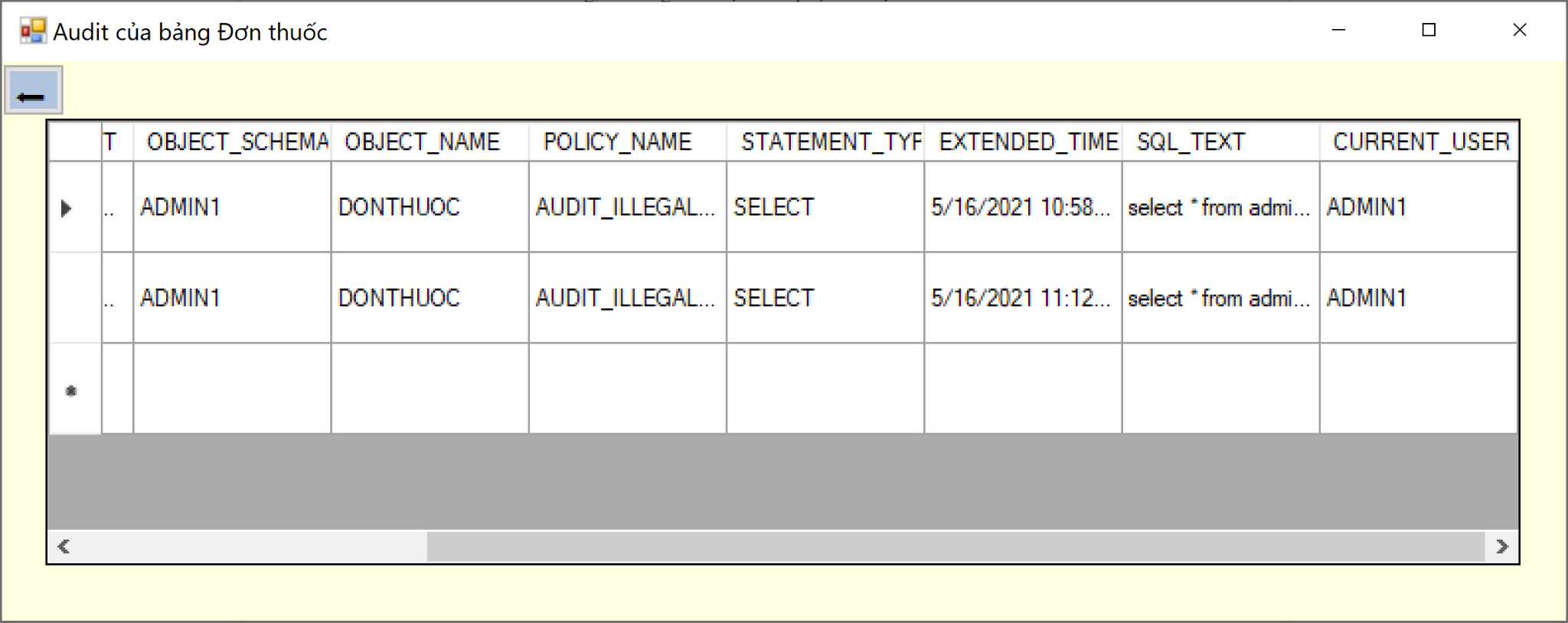
=> Chính vì vậy, nhu cầu cần thiết ghi nhận lại việc truy cập thông tin trái phép để làm bằng chứng, dễ dàng cho việc khôi phục dữ liệu là rất cần thiết.

- Lý do sử dụng: Audit là cơ chế sẽ giúp ghi nhận lại các hành động của các user trong hệ thống khi họ đang làm một hành động không được cho phép bởi admin của hệ thống.

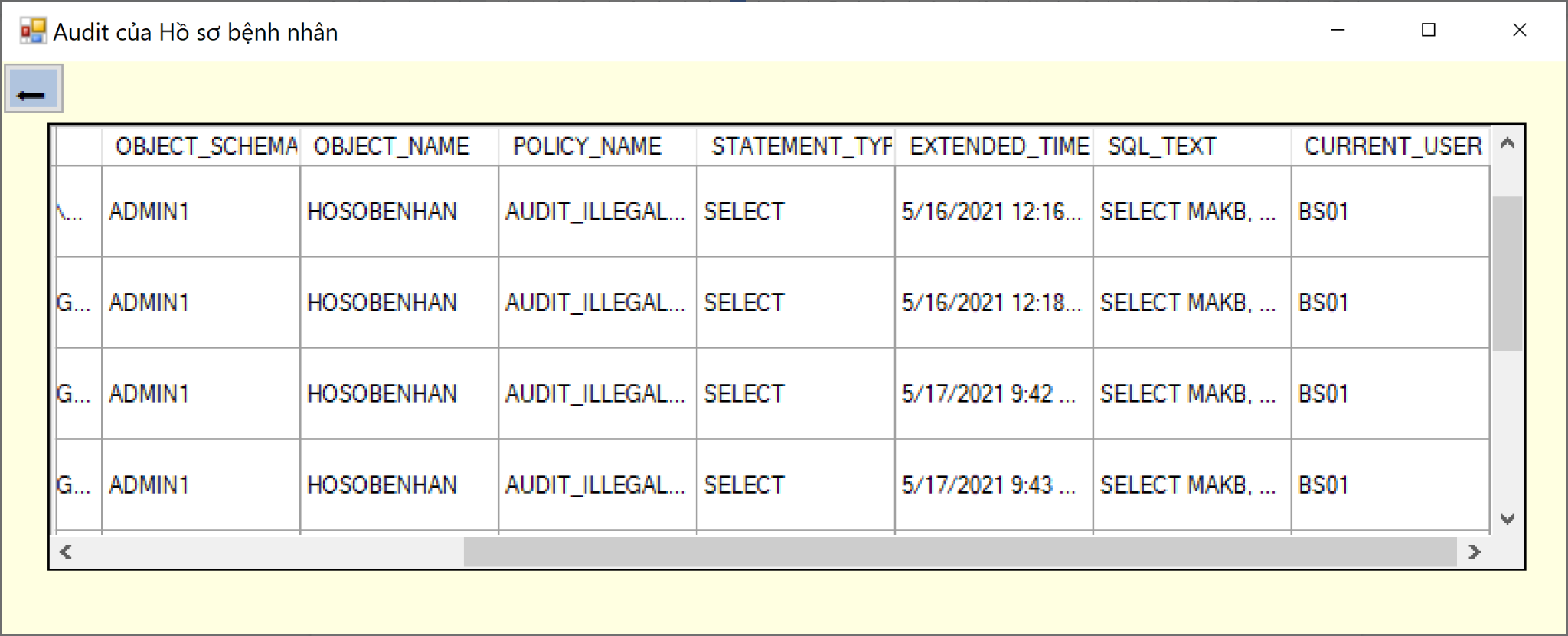
* Thuyết minh kết quả đạt được:
* Audit việc đăng nhập của người dùng:



* Audit của truy cập bảng Đơn thuốc:



* Audit của truy cập bảng Hồ sơ bệnh nhân:



# TÀI LIỆU THAM KHẢO

* [Oracle column-level security (dba-oracle.com)](http://dba-oracle.com/t_column_level_security.htm)
* Slide bộ môn An toàn và bảo mật hệ thống thông tin - Moodle