TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

-----o0o-----

BÀI TẬP USER AUTHENTICATION REPORT

Môn An Toàn Và Bảo Mật Dữ Liệu Trong Hệ Thống Thông Tin

Giảng viên: Phạm Thị Bạch Huệ

Lương Vĩ Minh

Tiết Gia Hồng

Trưởng nhóm **20H3T-04**: 20120624 – Mai Quyết Vang

Thành viên:

20120466 – Trần Thị Thu Hà

20120592 – Lê Minh Tiến

20120595 – Phạm Minh Tiến

Thành phố Hồ Chí Minh, ngày 30 tháng 6 năm 2023

MỤC LỤC

MỤC LỤC I
BẢNG PHÂN CÔNG II
PHẦN 2: THỰC HIỆN CÁC CHÍNH SÁCH BẢO MẬT 1
1. BÁO CÁO: LƯỢC ĐỒ CSDL CUỐI CÙNG ĐƯỢC CÀI ĐẶT, LIỆT KÊ CHÍNH SÁCH BẢO MẬT, PHÂN TÍCH VÀ PHÂN LOẠI, ĐỀ RA GIẢI PHÁP CÀI ĐẶT. 20% 1
1.1. Lược đồ CSDL cuối cùng được cài đặt 1
1.2. Chính sách bảo mật 3
1.3. Phân tích và phân loại 3
1.4. Giải pháp cài đặt 4
2. HOÀN THIỆN CÀI ĐẶT CÁC CHÍNH SÁCH BẢO MẬT NÊU TRONG ĐỒ ÁN, GỒM CÀI ĐẶT MỨC CƠ SỞ DỮ LIỆU VÀ CUNG CẤP GIAO DIỆN MINH HỌA CHO TỪNG VAI TRÒ NGƯỜI DÙNG (DAC, RBAC, VPD, MAC). 50% ..4
2.1. Chính sách 1: Nhân viên 5
2.2. Chính sách 2: Quản lý trực tiếp 9
2.3. Chính sách 3: Trưởng phòng 12
2.4. Chính sách 4: Tài chính 16
2.5. Chính sách 5: Nhân sự 22
2.6. Chính sách 6: Trưởng đề án 27
3. MÃ HÓA (CHỈ CẦN CÀI ĐẶT 1 CHÍNH SÁCH, TRÌNH BÀY RÕ PP QUẢN LÝ KHÓA, TRAO ĐỔI KHÓA, THAY ĐỔI KHÓA) VÀ GIAO DIỆN. 30
3.1. User vai trò nào sẽ thực hiện mã hóa? 30
3.2. Mã hóa dữ liệu ở mức nào? Vì sao chọn mức mã hóa đề nghị? 30
3.3. Có cần thay đổi gì về cấu trúc lưu trữ dữ liệu hay không? 31
3.4. Các khía cạnh của cơ chế quản lý khóa đề nghị 32
3.4.1. Thiết lập khóa: 32
3.4.2. Lưu trữ khóa: 32
3.4.3. Phân phối khóa: 33
4. OLS 34
5. AUDIT CƠ BẢN VÀ FGA (4 CHÍNH SÁCH). (HÀ) 35
5.1. Audit cơ bản 36
5.2. Fine – grained Auditing (FGA) 36
5.3. Các chính sách 37
TÀI LIỆU THAM KHẢO 41

DANH SÁCH CHỨC NĂNG ĐÃ HOÀN TẤT

STT	Nội dung	Hoàn tất
1	**PHẦN 1: HỆ THỐNG DÀNH CHO NGƯỜI QUẢN TRỊ BẢO MẬT** **Yêu cầu: Xây dựng giao diện cho phép người quản trị**	x
1.1	Xem danh sách các đối tượng hiện có trên CSDL (user, role, table, view, ….)	x
1.2	Thêm mới đối tượng (user, role)	x
1.3	Phân quyền/ lấy lại quyền của một user/ role.	x
1.4	Xem quyền của một chủ thể cụ thể.	x
2	**PHẦN 2: HIỆN THỰC CÁC CHÍNH SÁCH BẢO MẬT**	x
2.1	Báo cáo: Lược đồ CSDL cuối cùng được cài đặt, liệt kê chính sách bảo mật, phân tích và phân loại, đề ra giải pháp cài đặt.	x
	Hoàn thiện cài đặt các chính sách bảo mật nêu trong đồ án, gồm cài đặt mức cơ sở dữ liệu và cung cấp giao diện minh họa cho từng vai trò người dùng (DAC, RBAC, MAC).	x
2.2	Mã hóa (chỉ cần cài đặt 1 chính sách, trình bày rõ PP quản lý khóa, trao đổi khóa, thay đổi khóa) và giao diện.	x
2.3	Audit cơ bản và FGA (4 chính sách).	x

BẢNG PHÂN CÔNG

MSSV	Họ tên	Công việc	Mức độ hành thành
20120466	Trần Thị Thu Hà	Code giao diện CS#5, yêu cầu 3 (OLS), yêu cầu 4 (audit)	90%
20120592	Lê Minh Tiến	Code giao diện CS#1, CS#6, hỗ trợ mã hóa, yêu cầu 1 (lược đồ CSDL cuối cùng).	90%
20120595	Phạm Minh Tiến	Code giao diện CS#4, phụ trách mã hóa, yêu cầu 1.	90%
20120624	Mai Quyết Vang	Code giao diện CS#2, CS#3, hỗ trợ yêu cầu 3, yêu cầu 4.	90%

PHẦN 2: THỰC HIỆN CÁC CHÍNH SÁCH BẢO MẬT

1. Báo cáo: Lược đồ CSDL cuối cùng được cài đặt, liệt kê chính sách bảo mật, phân tích và phân loại, đề ra giải pháp cài đặt. 20%

1.1. Lược đồ CSDL cuối cùng được cài đặt

Ở phần lược đồ CSDL này, nhóm chúng em không thay đổi trên các quan hệ có sẵn, mà chỉ thêm quan hệ THUONG.

NHANVIEN (MANV, TENNV, PHAI, NGAYSINH, DIACHI, SODT, LUONG, PHUCAP, VAITRO, MANQL, PHG)

```
create table NHANVIEN
(
    MANV    varchar2(5),
    TENNV   varchar2(30),
    PHAI    varchar2(5),
    NGAYSINH    date,
    DIACHI  varchar2(50),
    SODT number(10),
    LUONG   varchar2(4000),
    PHUCAP varchar2(4000),
    VAITRO varchar2(20),
    MANQL   varchar2(5),
    PHG     varchar2(5),
    CONSTRAINT pk_nv primary key (MANV)
)TABLESPACE DA_ATBM;
/
```

PHONGBAN (MAPB, TENPB, TRPHG)

```
create table PHONGBAN
(
    MAPHG   varchar2(5),
    TENPHG  varchar2(30),
    TRPHG   varchar2(5),
    CONSTRAINT pk_pb primary key (MAPHG)
)TABLESPACE DA_ATBM;
/
```

DEAN (MADA, TENDA, NGAYBD, PHONG)

```
create table DEAN
(
    MADA varchar2(5),
    TENDA varchar2(50),
    NGAYBD date,
    PHONG varchar2(5),
    CONSTRAINT pk_da primary key (MADA)
)TABLESPACE DA_ATBM;
/
```

PHANCONG (MANV, MADA, THOIGIAN)

```
create table PHANCONG
(
    MANV    varchar2(5),
    MADA    varchar2(5),
    THOIGIAN    date,
    CONSTRAINT pk_pc primary key (MANV, MADA)
)TABLESPACE DA_ATBM;
/
```

THUONG (MANV, DIPTHUONG, TIENTHUONG)

```
CREATE TABLE THUONG (
    MANV VARCHAR2(5) PRIMARY KEY,
    DIPTHUONG VARCHAR2(4000),
    TIENTHUONG VARCHAR2(4000)
)TABLESPACE DA_ATBM;
/
```

Trong quan hệ THUONG:

Thay đổi tên bảng từ "COUPLE_OF_KEYS" (như bảng ở ảnh bên dưới) thành "THUONG" (THƯỞNG) là **để che dấu ý nghĩa thực sự của các cột** và ngăn người khác biết chi tiết về cấu trúc dữ liệu. Những người không có quyền truy cập dữ liệu, hay không phải là người thiết kế cơ sở dữ liệu sẽ không biết mục đích thực sự của bảng "THUONG" này là phục vụ cho việc mã hóa- giải mã, mà chỉ có thể hiểu đây là bảng dùng để lưu lại THƯỞNG của các nhân viên. Điều này có thể giúp ngăn chặn một số cuộc tấn công hoặc nỗ lực không mong muốn của người khác đối với dữ liệu.

"THUONG" bản chất là một bảng dùng để lưu cặp khóa public-private dùng để mã hóa và giải mã thuộc tính LUONG và PHUCAP của mỗi nhân viên. 2 thuộc tính DIPTHUONG,

TIENTHUONG tương đương với 2 thuộc tính public_key, private_key là những chuỗi được mã hóa.

```
CREATE TABLE COUPLE_OF_KEYS (
    MANV VARCHAR2(100) PRIMARY KEY,
    public_key VARCHAR2(4000),
    private_key VARCHAR2(4000)
);
/
```

1.2. Chính sách bảo mật

- Sử dụng DAC (Discretionary Access Control) kết hợp với RBAC (Role-Based Access Control): Chia người dùng thành 6 ROLE khác nhau để quản lý quyền truy cập dữ liệu. Sử dụng các kỹ thuật view để giới hạn truy cập dữ liệu cho từng vai trò một cách phù hợp với thực tế của từng ROLE.
- Sử dụng kỹ thuật mã hóa dữ liệu nhạy cảm: Áp dụng mã hóa cho các thuộc tính nhạy cảm như Lương và Phụ cấp để bảo vệ thông tin cá nhân. Sử dụng cặp khóa public-private trong bảng "THUONG" để mã hóa và giải mã các thuộc tính này.
- Sử dụng MAC/OLS (Mandatory Access Control/Object Labeling System): Áp dụng phân chia dữ liệu phát tán thông qua việc gán nhãn và quản lý quyền truy cập dựa trên nhãn của đối tượng.
- Sử dụng Audit (giao dịch kiểm tra) để ghi nhận và kiểm tra các vấn đề liên quan đến bảo mật và truy cập dữ liệu.

1.3. Phân tích và phân loại

- Sử dụng DAC kết hợp RBAC:

  Phân tích: Chính sách này nhằm chia người dùng thành 6 ROLE khác nhau để quản lý quyền truy cập dữ liệu. Sử dụng view để giới hạn truy cập cho từng vai trò.

  Phân loại: Đây là chính sách kiểm soát truy cập dựa trên quyền và vai trò người dùng.

- Sử dụng mã hóa dữ liệu nhạy cảm:

  Phân tích: Chính sách này nhằm bảo vệ thông tin cá nhân bằng cách áp dụng mã hóa cho các thuộc tính nhạy cảm như Lương và Phụ cấp.

Phân loại: Đây là chính sách bảo mật dữ liệu.

- Sử dụng MAC/OLS:

  Phân tích: Chính sách này nhằm áp dụng phân chia dữ liệu phát tán thông qua việc gán nhãn và quản lý quyền truy cập dựa trên nhãn của đối tượng.

  Phân loại: Đây là chính sách kiểm soát truy cập dựa trên nhãn (label-based).

- Sử dụng Audit:

  Phân tích: Chính sách này nhằm ghi nhận và kiểm tra các vấn đề liên quan đến bảo mật và truy cập dữ liệu.

  Phân loại: Đây là chính sách giám sát và kiểm tra bảo mật.

1.4. Giải pháp cài đặt

- Xây dựng hệ thống xác thực: Sử dụng tài khoản người dùng và mật khẩu để xác thực quyền truy cập vào cơ sở dữ liệu.
- Áp dụng mã hóa dữ liệu: Sử dụng thuật toán mã hóa RSA trên mức ứng dụng để mã hóa và giải mã lương và phụ cấp với các khóa public-private trong bảng "THUONG" và chỉ cho phép các người dùng có quyền truy cập được giải mã và mã hóa dữ liệu.
- Thiết lập kiểm soát truy cập: Xác định quyền truy cập cho từng người dùng và vai trò, đảm bảo rằng họ chỉ có thể truy cập và sửa đổi dữ liệu cần thiết.
- Thiết lập hệ thống ghi nhận giao dịch kiểm tra (audit) để ghi lại các hoạt động liên quan đến bảo mật và truy cập dữ liệu. Đảm bảo việc kiểm tra định kỳ các ghi chú audit để phát hiện các hành vi không mong muốn và nâng cao quản lý bảo mật.

2. Hoàn thiện cài đặt các chính sách bảo mật nêu trong đồ án, gồm cài đặt mức cơ sở dữ liệu và cung cấp giao diện minh họa cho từng vai trò người dùng (DAC, RBAC, VPD, MAC). 50%

Tạo 6 role: NHANVIEN, QLTRUCTIEP, TRUONGPHONG, NHANSU, TAICHINH, TRUONGDEAN.

Tất các các USERS được tạo đều được gán role NHANVIEN, và ứng với từng vai trò, các USERS sẽ được gán thêm các role tương ứng.

2.1. Chính sách 1: Nhân viên

Yêu cầu:

- Có quyền xem tất cả các thuộc tính trên quan hệ NHANVIEN và PHANCONG liên quan
- đến chính nhân viên đó.
- Có thể sửa trên các thuộc tính NGAYSINH, DIACHI, SODT liên quan đến chính nhân viên đó.
- Có thể xem dữ liệu của toàn bộ quan hệ PHONGBAN và DEAN.

Ý tưởng

Sử dụng RBAC. Tạo ra role NHANVIEN, sau đó tạo ra các view, proceduce mà đảm bảo các chính sách bảo mật của một NHANVIEN. Tiếp đó grant role này đến tất cả các USER, bởi vì mọi người dùng đều có các quyền của một NHANVIEN. Khi đó mọi USER được tạo ra đều có các quyền của 1 NHANVIEN.

Chính sách	Đối tượng	Quyền
- Xem tất cả thuộc tính trên quan hệ NHANVIEN liên quan đến chính nhân viên đó	(VIEW) NV_XemThongTinChinhMinh	SELECT
- Xem tất cả thuộc tính trên quan hệ PHANCONG liên quan đến chính nhân viên đó	(VIEW) NV_XemThongTinPhanCong	SELECT
- Có thể sửa trên các thuộc tính NGAYSINH, DIACHI, SODT liên quan đến chính nhân viên đó	(PROCEDURE) NV_SUATHONGTIN	EXECUTE

- Xem dữ liệu toàn bộ PHONGBAN	(VIEW) NV_XemThongTinPhongBan	SELECT
- Xem dữ liệu toàn bộ DEAN	(VIEW) NV_XemThongTinDeAn	SELECT

Mức cơ sở dữ liệu:

VIEW: NV_XemThongTinChinhMinh

```
--NhanVien quyền 1: xem thông tin cá nhân của chính mình
create or replace view NV_XemThongTinChinhMinh
as
    select* from NhanVien
    where MaNV= SYS_CONTEXT('USERENV', 'SESSION_USER');
/
```

VIEW: NV_XemThongTinPhanCong

```
--NhanVien quyen 2: xem thong tin phan cong cua chinh minh
create or replace view NV_XemThongTinPhanCong
as
    select* from PhanCong
    where MaNV= SYS_CONTEXT('USERENV', 'SESSION_USER');
/
```

PROCEDURE: NV_SUATHONGTIN

```
--Nhan vien update thong tin NGAYSINH, DIACHI,SODT cua chinh minh
CREATE OR REPLACE PROCEDURE NV_SUATHONGTIN(
    NGAYSINH_ IN DATE,
    DIACHI_ IN VARCHAR2,
    SODT_ IN NUMBER
)
IS
BEGIN
    UPDATE ATBM_ADMIN.NV_XemThongTinChinhMinh
    SET  NGAYSINH=NGAYSINH_,DIACHI=DIACHI_,SODT=SODT_
    WHERE MANV = SYS_CONTEXT('USERENV', 'SESSION_USER') ;
    COMMIT;
END;
/
```

VIEW: NV_XemThongTinPhongBan

```
--NhanVien quyen 4: xem tat ca phong ban
create or replace view NV_XemThongTinPhongBan
as
    select* from PhongBan;
/
```

VIEW: NV_XemThongTinDeAn

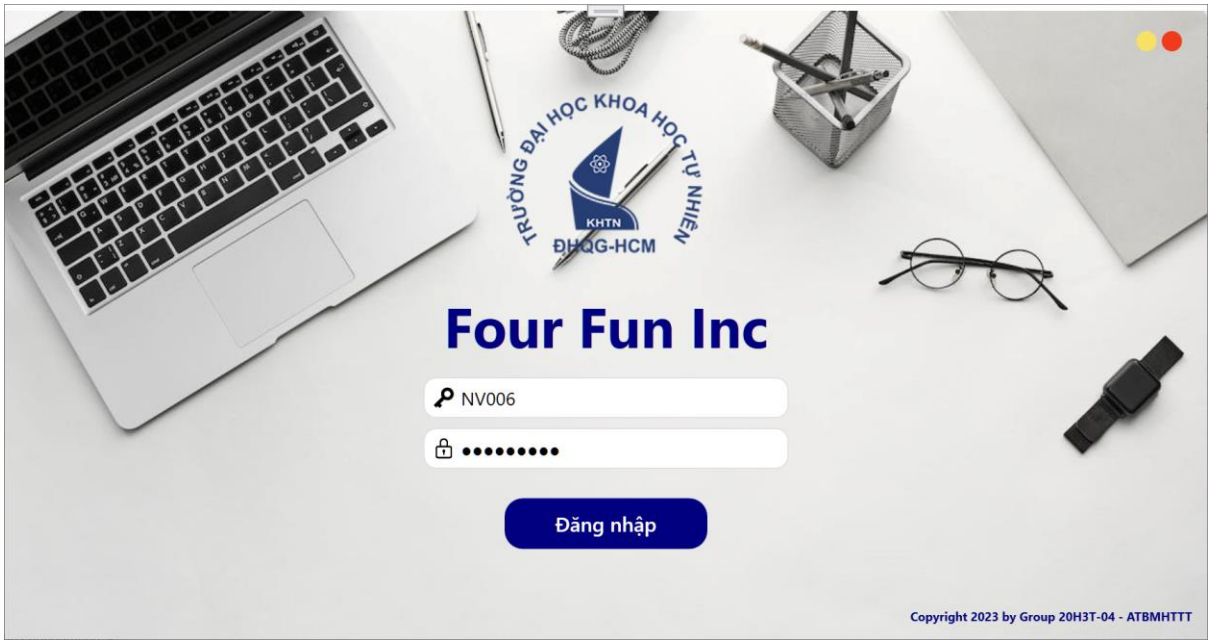
```
--NhanVien quyen 5: xem tat ca de an
create or replace view NV_XemThongTinDeAn
as
    select* from DeAn;
/
```

GRANT:

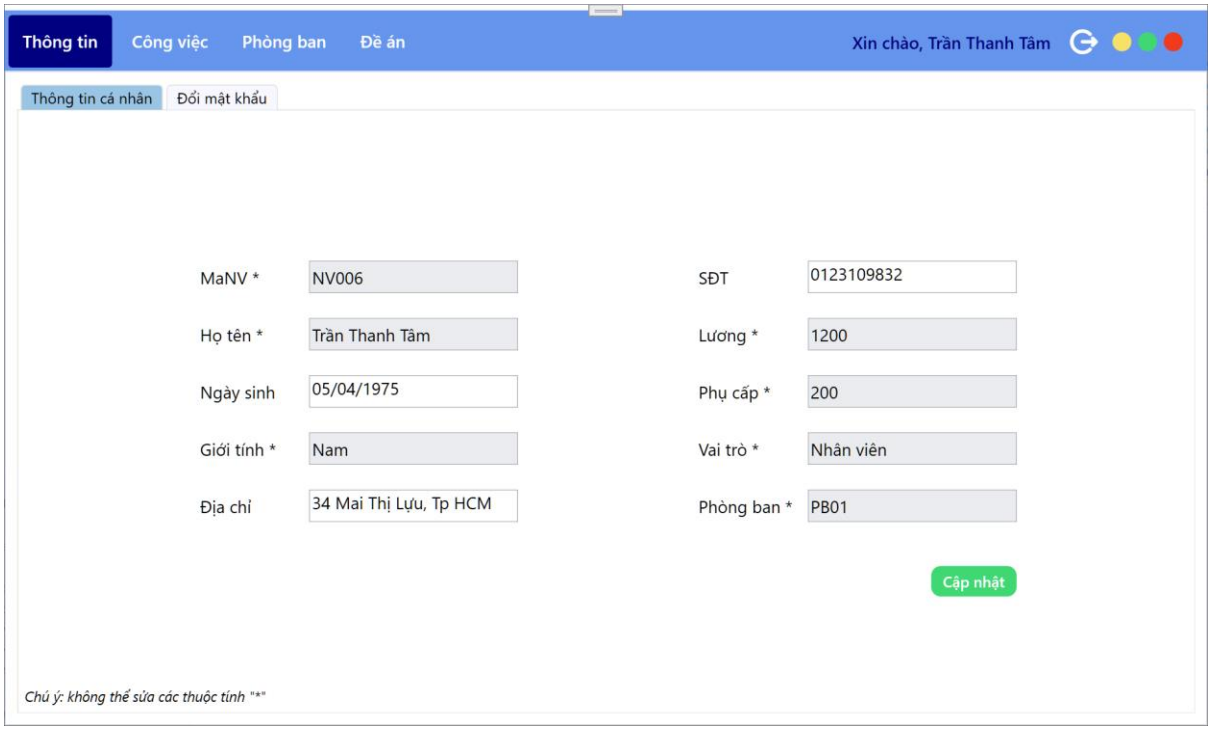
```
--Grant cac quyen cho role NHANVIEN
grant select On NV_XemThongTinChinhMinh to NhanVien;
grant select On NV_XemThongTinPhanCong to NhanVien;
grant select On NV_XemThongTinPhongBan to NhanVien;
grant select On NV_XemThongTinDeAn to NhanVien;
grant execute On NV_SUATHONGTIN to NhanVien;
/
```

Mức giao diện:

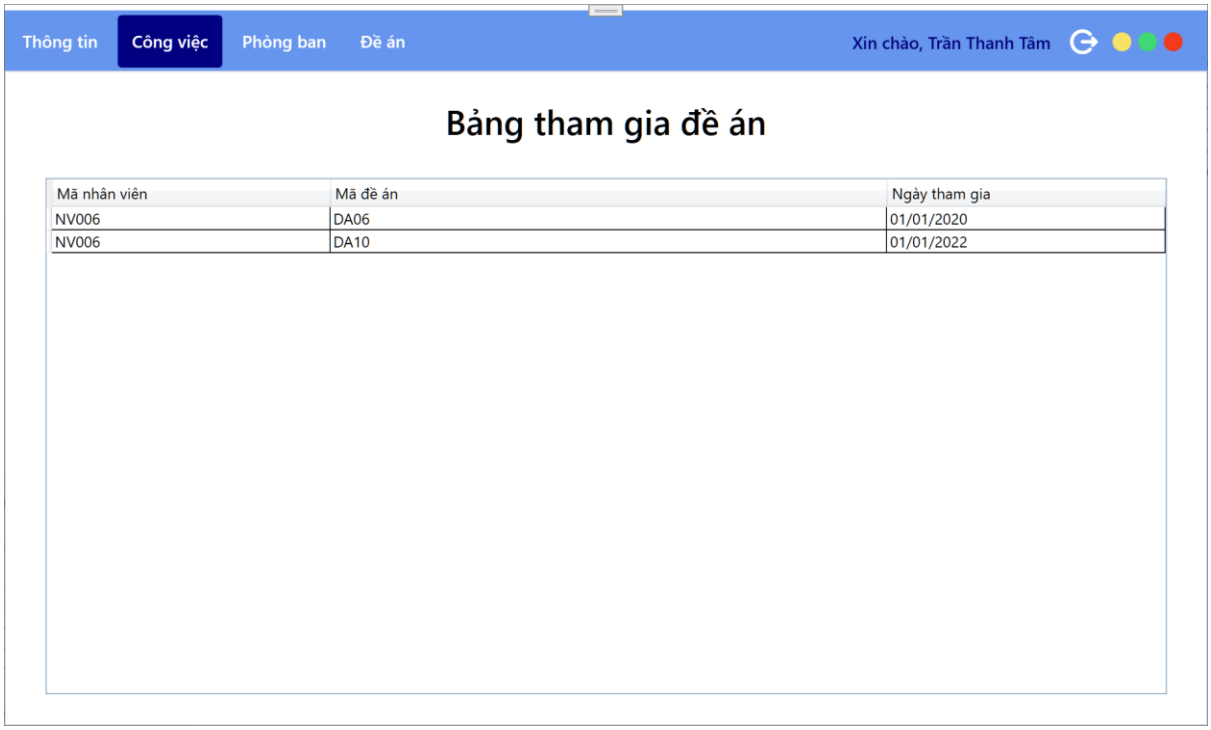
Màn hình đăng nhập của “Nhân viên”:



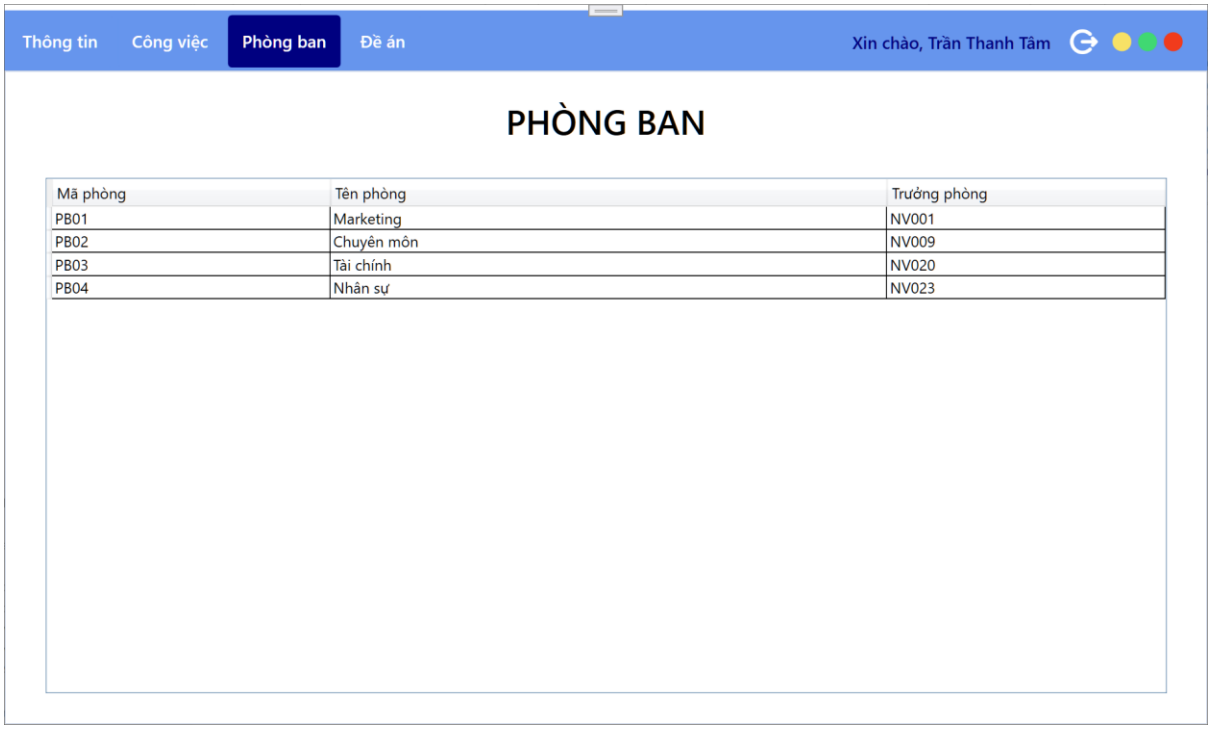
"Nhân viên" xem thông tin của chính mình, tại đây nhân viên có thể sửa các thuộc tính như "Ngày sinh", "Địa chỉ", "SĐT":



"Nhân viên" xem PHANCONG của chính mình:



"Nhân viên" xem tất cả phòng ban:



Mã phòng	Tên phòng	Trưởng phòng
PB01	Marketing	NV001
PB02	Chuyên môn	NV009
PB03	Tài chính	NV020
PB04	Nhân sự	NV023

"Nhân viên" xem tất cả đề án:

Mã đề án	Tên đề án	Ngày bắt đầu	Phòng ban phụ trách
DA01	Thiết kế mạng ABC company	01/01/2016	PB02
DA02	Thử nghiệm cáp quang CQ1	01/01/2018	PB02
DA03	Tuyển nhân sự tháng 1_2019	01/01/2019	PB04
DA04	Thử nghiệm hệ thống M1 lần 1	01/06/2019	PB02
DA05	Lắp đặt cáp quang CQ1	01/12/2019	PB02
DA06	Workshop hệ thống mạng	01/01/2020	PB01
DA07	Tuyển nhân sự tháng 6_2020	01/06/2020	PB04
DA08	Thử nghiệm hệ thống M1 lần 2	01/08/2020	PB02
DA09	Marketing hệ thống M1	01/01/2021	PB01
DA10	Training nhân viên 1_2022	01/01/2022	PB04

2.2. Chính sách 2: Quản lý trực tiếp

Yêu cầu:

- Q có quyền như là một nhân viên thông thường (vai trò "Nhân viên"). Ngoài ra, với các dòng dữ liệu trong quan hệ NHANVIEN liên quan đến các nhân viên N mà Q

quản lý trực tiếp thì Q được xem tất cả các thuộc tính, trừ thuộc tính LUONG và PHUCAP.

– Có thể xem các dòng trong quan hệ PHANCONG liên quan đến chính Q và các nhân viên N được quản lý trực tiếp bởi Q.

Ý tưởng:

Cài đặt view sử dụng decode để bảo mật cột lương và phụ cấp với bảng NHANVIEN. Cùng với đó là tạo view cho bảng PHANCONG để chỉ có thể xem phân công liên quan đến nhân viên mình quản lý. Sau đó gắn role này cho các user có vai trò là QL trực tiếp.

Chính sách	Đối tượng	Quyền
- Xem tất cả thuộc tính trên quan hệ NHANVIEN trừ cột LUONG, PHUCAP liên quan đến Q và các nhân viên N mà Q quản lý trực tiếp	(VIEW) QL_XEMNHANVIEN	SELECT
- Xem tất cả thuộc tính trên quan hệ PHANCONG liên quan đến chính Q và các nhân viên N được quản lý trực tiếp bởi Q.	(VIEW) QL_XEMPHANCONG	SELECT

Mức cơ sở dữ liệu:

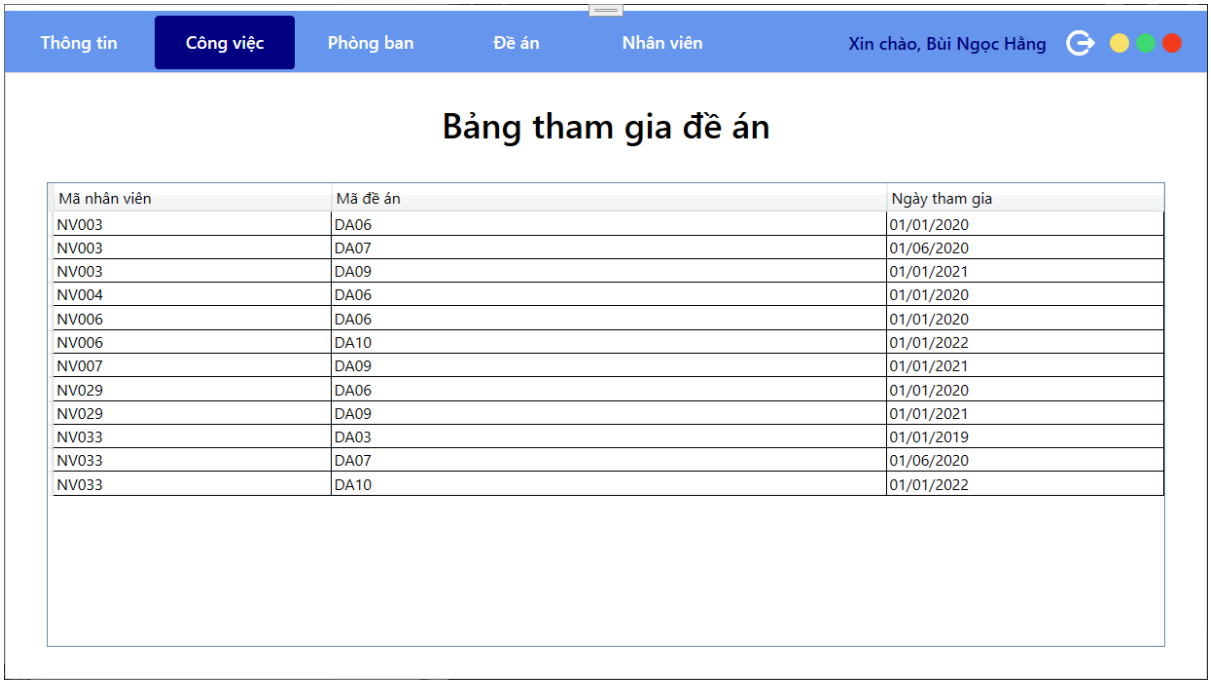
```
create or replace view QL_XEMNHANVIEN
as
    select  MANV, TENNV,PHAI,NGAYSINH,DIACHI,
    SODT, DECODE(MANV,sys_context('USERENV', 'CURRENT_USER'),luong,NULL) LUONG ,
    DECODE(MANV,sys_context('USERENV', 'CURRENT_USER'),PHUCAP,NULL) PHUCAP,
    VAITRO ,MANQL ,PHG
    from nhanvien
    where MANV = sys_context('USERENV', 'CURRENT_USER')
        or MANQL = (select MANV
                    from nhanvien
                    where MANV = sys_context('USERENV', 'CURRENT_USER'));
/
```

```
--được quản lý trực tiếp bởi Q.
create or replace view QL_XEMPHANCONG
as
    select  MANV,MADA,THOIGIAN
    from PHANCONG
    where MANV = sys_context('USERENV', 'CURRENT_USER')
        or MANV in (select MANV
                    from nhanvien
                    where MANQL = sys_context('USERENV', 'CURRENT_USER'));
/
--alter session set "_ORACLE_SCRIPT"=true;
--GAN VIEW CHO ROLE
GRANT SELECT ON QL_XEMNHANVIEN TO QLTRUCTIEP;
GRANT SELECT ON QL_XEMPHANCONG TO QLTRUCTIEP;
```

Mức giao diện:

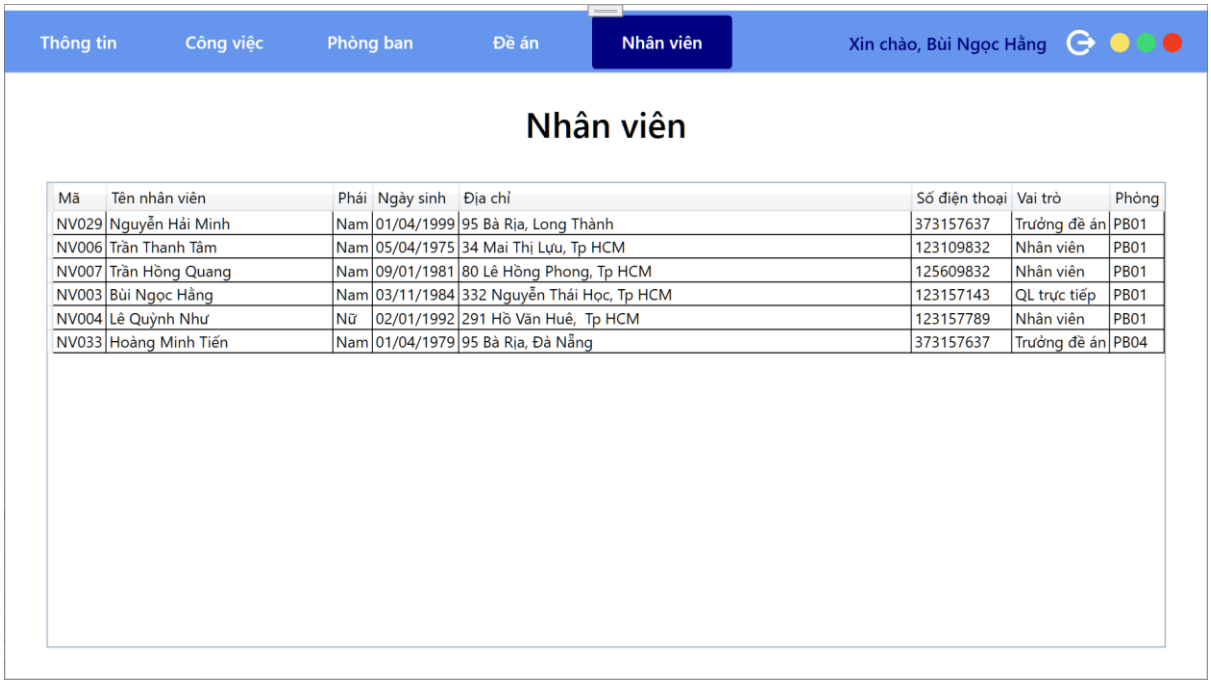
Ngoài các giao diện giống như một "Nhân viên", "QL trực tiếp" còn có các màn hình giao diện khác.

Giống với "Nhân viên", "QL trực tiếp" có thể coi được PHANCONG của mình, ngoài ra còn coi được PHANCONG của các "Nhân viên" mà mình quản lý trực tiếp:



Mã nhân viên	Mã đề án	Ngày tham gia
NV003	DA06	01/01/2020
NV003	DA07	01/06/2020
NV003	DA09	01/01/2021
NV004	DA06	01/01/2020
NV006	DA06	01/01/2020
NV006	DA10	01/01/2022
NV007	DA09	01/01/2021
NV029	DA06	01/01/2020
NV029	DA09	01/01/2021
NV033	DA03	01/01/2019
NV033	DA07	01/06/2020
NV033	DA10	01/01/2022

"QL trực tiếp" xem thông tin (trừ lương, phụ cấp) của các nhân viên mà mình quản lý:



Mã	Tên nhân viên	Phái	Ngày sinh	Địa chỉ	Số điện thoại	Vai trò	Phòng
NV029	Nguyễn Hải Minh	Nam	01/04/1999	95 Bà Rịa, Long Thành	373157637	Trưởng đề án	PB01
NV006	Trần Thanh Tâm	Nam	05/04/1975	34 Mai Thị Lựu, Tp HCM	123109832	Nhân viên	PB01
NV007	Trần Hồng Quang	Nam	09/01/1981	80 Lê Hồng Phong, Tp HCM	125609832	Nhân viên	PB01
NV003	Bùi Ngọc Hằng	Nam	03/11/1984	332 Nguyễn Thái Học, Tp HCM	123157143	QL trực tiếp	PB01
NV004	Lê Quỳnh Như	Nữ	02/01/1992	291 Hồ Văn Huê, Tp HCM	123157789	Nhân viên	PB01
NV033	Hoàng Minh Tiến	Nam	01/04/1979	95 Bà Rịa, Đà Nẵng	373157637	Trưởng đề án	PB04

2.3. Chính sách 3: Trưởng phòng

Yêu cầu

- T có quyền như là một nhân viên thông thường (vai trò "Nhân viên"). Ngoài ra, với các dòng trong quan hệ NHANVIEN liên quan đến các nhân viên thuộc phòng ban mà T làm trưởng phòng thì T có quyền xem tất cả các thuộc tính, trừ thuộc tính LUONG và PHUCAP.
- Có thể thêm, xóa, cập nhật trên quan hệ PHANCONG liên quan đến các nhân viên thuộc phòng ban mà T làm trưởng phòng.

Chính sách	Đối tượng	Quyền
- Xem tất cả thuộc tính trên quan hệ NHANVIEN liên quan đến chính nhân viên đó, và các nhân viên quản lý bởi trưởng phòng đó nhưng trừ Lương và Phụ cấp	(VIEW) TP_NHANVIEN	SELECT

- Có thể thêm công việc cho nhân viên được quản lý bởi chính mình	(PROCEDURE) TP_ThemPhanCong	EXECUTE
- Có thể sửa công việc cho nhân viên được quản lý bởi chính mình	(PROCEDURE) TP_SuaPhanCong	EXECUTE
- Có thể xoá công việc cho nhân viên được quản lý bởi chính mình	(PROCEDURE) TP_XoaPhanCong	EXECUTE

Mức cơ sở dữ liệu:

Ý tưởng: cấp quyền sử dụng dữ liệu trên quan hệ NHANVIEN. Tạo view và decode để che các thông tin không cần thiết.

```
---------------------- Chính sách #3 Trưởng Phòng ----------------------
-- T có quyền như là một nhân viên thông thường (vai trò “Nhân viên”). Ngoài ra,
CREATE OR REPLACE VIEW TP_NHANVIEN AS
SELECT  MANV, TENNV, PHAI,
        NGAYSINH, DIACHI, SODT,
        DECODE(MANV,SYS_CONTEXT('USERENV', 'SESSION_USER'),LUONG,NULL) LUONG,
        DECODE(MANV,SYS_CONTEXT('USERENV', 'SESSION_USER'),PHUCAP,NULL) PHUCAP,
        VAITRO, MANQL, PHG
FROM NHANVIEN
WHERE PHG = (select PHG from NHANVIEN
             where MANV =  SYS_CONTEXT('USERENV', 'SESSION_USER'));
/
```

Procedure TP_ThemPhanCong dùng để thêm công việc cho nhân viên vào từng đề án với thời gian quy định.

```
CREATE OR REPLACE PROCEDURE TP_ThemPhanCong(
    MaNV_ IN VARCHAR2,
    MaDA_ IN VARCHAR2,
    ThoiGian_ IN DATE
)
IS
BEGIN
    INSERT INTO ATBM_ADMIN.TP_PHANCONG (MANV, MADA, THOIGIAN)
    VALUES (MaNV_, MaDA_, ThoiGian_);
    COMMIT;
END;
/
```

Procedure TP_SuaPhanCong dùng để chỉnh sửa thời gian công việc cho nhân viên ứng với từng đề án cụ thể.

```
CREATE OR REPLACE PROCEDURE TP_SuaPhanCong(
    MaNV_ IN VARCHAR2,
    MaDA_ IN VARCHAR2,
    ThoiGian_ IN DATE
)
IS
BEGIN
    UPDATE ATBM_ADMIN.TP_PHANCONG
    SET THOIGIAN = ThoiGian_
    WHERE MANV = MaNV_ AND MADA = MaDA_;
    COMMIT;
END;
/
```

Procedure TP_ThemPhanCong dùng để xoá công việc cho nhân viên ứng với đề án cụ thể.

```
create or replace PROCEDURE TP_XoaPhanCong(
    MaNV_ in varchar2,
    MaDA_ in VARCHAR2
)
IS
BEGIN
    delete from ATBM_ADMIN.TP_PHANCONG
    where MANV = MaNV_ and MADA = MaDA_;
END;
```

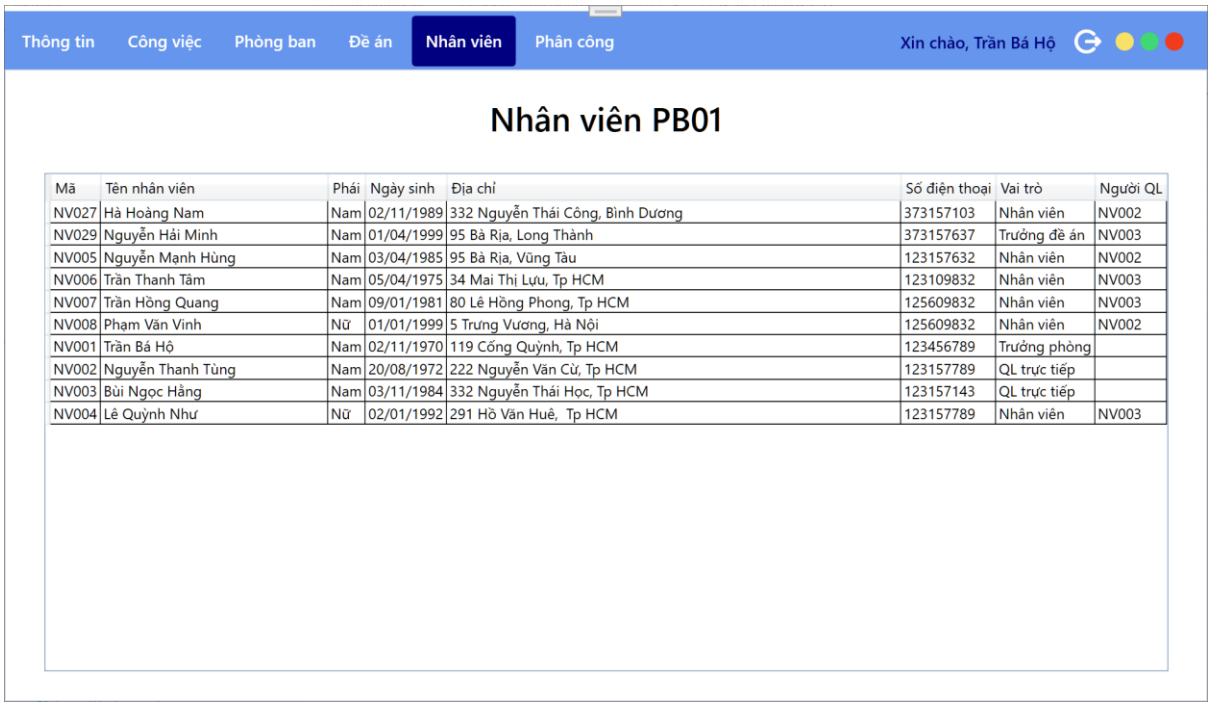
Gán các quyền procedure của trưởng phòng cho role TRUONGPHONG.

```
grant EXECUTE ON TP_ThemPhanCong to TRUONGPHONG;
grant EXECUTE ON TP_SuaPhanCong to TRUONGPHONG;
grant EXECUTE ON TP_XoaPhanCong to TRUONGPHONG;
/
```

Mức giao diện:

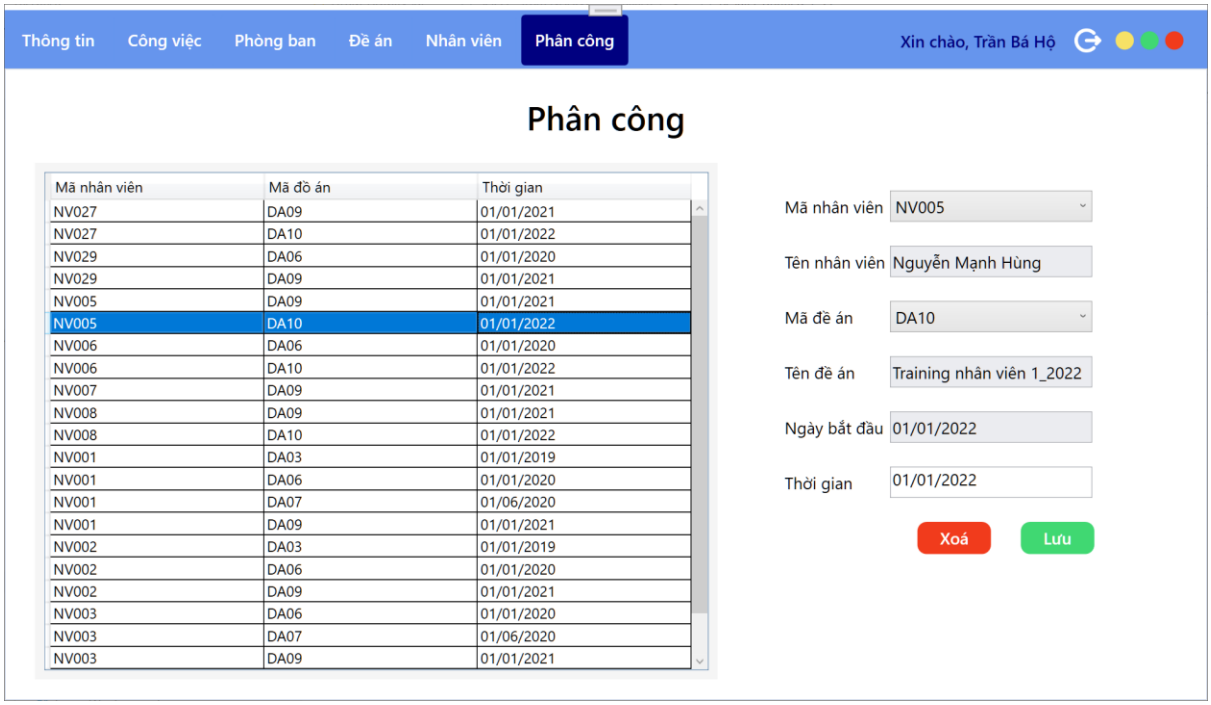
Ngoài các giao diện giống như một "Nhân viên", "Trưởng phòng" còn có các màn hình giao diện khác.

Xem thông tin (trừ lương, phụ cấp) "Nhân viên" thuộc phòng ban mà mình làm trưởng phòng:



Mã	Tên nhân viên	Phái	Ngày sinh	Địa chỉ	Số điện thoại	Vai trò	Người QL
NV027	Hà Hoàng Nam	Nam	02/11/1989	332 Nguyễn Thái Công, Bình Dương	373157103	Nhân viên	NV002
NV029	Nguyễn Hải Minh	Nam	01/04/1999	95 Bà Rịa, Long Thành	373157637	Trưởng đề án	NV003
NV005	Nguyễn Mạnh Hùng	Nam	03/04/1985	95 Bà Rịa, Vũng Tàu	123157632	Nhân viên	NV002
NV006	Trần Thanh Tâm	Nam	05/04/1975	34 Mai Thị Lựu, Tp HCM	123109832	Nhân viên	NV003
NV007	Trần Hồng Quang	Nam	09/01/1981	80 Lê Hồng Phong, Tp HCM	125609832	Nhân viên	NV003
NV008	Phạm Văn Vinh	Nữ	01/01/1999	5 Trưng Vương, Hà Nội	125609832	Nhân viên	NV002
NV001	Trần Bá Hộ	Nam	02/11/1970	119 Cống Quỳnh, Tp HCM	123456789	Trưởng phòng	
NV002	Nguyễn Thanh Tùng	Nam	20/08/1972	222 Nguyễn Văn Cừ, Tp HCM	123157789	QL trực tiếp	
NV003	Bùi Ngọc Hằng	Nam	03/11/1984	332 Nguyễn Thái Học, Tp HCM	123157143	QL trực tiếp	
NV004	Lê Quỳnh Như	Nữ	02/01/1992	291 Hồ Văn Huê, Tp HCM	123157789	Nhân viên	NV003

Có thể thêm, xóa, cập nhật trên quan hệ PHANCONG liên quan đến các nhân viên thuộc phòng ban mà T làm trưởng phòng:



2.4. Chính sách 4: Tài chính

Yêu cầu:

- Có quyền như là một nhân viên thông thường (vai trò "Nhân viên").
- Xem trên toàn bộ quan hệ NHANVIEN và PHANCONG, có thể sửa trên thuộc tính LUONG và PHUCAP (thừa hành ban giám đốc).

Mức cơ sở dữ liệu:

Ý tưởng: cấp quyền sử dụng dữ liệu trên quan hệ NHANVIEN. Tạo view để xem toàn bộ quan hệ NHANVIEN và PHANCONG

Chính sách	Đối tượng	Quyền
Xem toàn bộ thông tin nhân viên	(VIEW) TC_XEMNHANVIEN	SELECT
Xem toàn bộ bảng PHANCONG	(VIEW) TC_XEMPHANCONG	SELECT
Cập nhật lương, phụ cấp trên bảng NHANVIEN	(PROCEDURE) TC_UPD_LUONG_PHUCAP	EXECUTE

```
CREATE OR REPLACE VIEW TC_XEMNHANVIEN AS
SELECT *
FROM ATBM_ADMIN.NHANVIEN ;
/
CREATE OR REPLACE VIEW TC_XEMPHANCONG AS
SELECT *
FROM ATBM_ADMIN.PHANCONG ;
/

GRANT SELECT ON TC_XEMNHANVIEN TO TAICHINH;
GRANT SELECT ON TC_XEMPHANCONG TO TAICHINH;
```

Tạo procedure và gán quyền để có thể sửa trên thuộc tính LUONG và PHUCAP

```
GRANT UPDATE (LUONG, PHUCAP) ON ATBM_ADMIN.TC_XEMNHANVIEN TO TAICHINH;
/
CREATE OR REPLACE PROCEDURE TC_UPD_LUONG_PHUCAP(
  p_manv IN VARCHAR2,
  LUONGMOI IN VARCHAR2,
  PHUCAPMOI IN VARCHAR2
) AS
  v_count NUMBER;
BEGIN
    -- Kiểm tra sự tồn tại của MANV trong bảng TC_XEMNHANVIEN
    SELECT COUNT(*) INTO v_count
    FROM ATBM_ADMIN.TC_XEMNHANVIEN
    WHERE MANV = p_manv;

    IF v_count = 0 THEN
        -- Thêm bản ghi mới nếu MANV không tồn tại
        INSERT INTO ATBM_ADMIN.TC_XEMNHANVIEN (MANV, LUONG, PHUCAP)
        VALUES (p_manv, LUONGMOI, PHUCAPMOI);
    ELSE
        -- Cập nhật LUONG và PHUCAP nếu MANV đã tồn tại
        UPDATE ATBM_ADMIN.TC_XEMNHANVIEN
        SET LUONG = LUONGMOI,
            PHUCAP = PHUCAPMOI
        WHERE MANV = p_manv;
    END IF;

    COMMIT;
END;
```

Ngoài ra, Tài chính còn có thể truy cập vào bảng khóa(tên giả là THUONG) để truy cập xem và chỉnh sửa khóa.Tạo chính sách VPD, chỉ cho Tài Chính xem toàn bộ thông tin trên bảng THUONG và nhân viên chỉ xem khóa của mình.

```
CREATE OR REPLACE FUNCTION keys_access_predicate (
    schema_name IN VARCHAR2,
    object_name IN VARCHAR2
) RETURN VARCHAR2
IS
    predicate VARCHAR2(4000);
    role_count NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO role_count
    FROM ATBM_ADMIN.TC_XEMNHANVIEN
    WHERE MANV = SYS_CONTEXT('USERENV', 'SESSION_USER')
    AND VAITRO = 'Tài chính';

    IF role_count = 1 THEN
        predicate := '1 = 1'; -- Cho phép truy cập toàn bộ thông tin cho vai trò TAICHINH
    ELSE
        predicate := 'MANV = SYS_CONTEXT(''USERENV'', ''SESSION_USER'')'; -- Chỉ cho phép truy cập thông tin của riêng người dùng
    END IF;

    RETURN predicate;
END;
/
```

```
DECLARE
    v_policy_exists NUMBER;
BEGIN
    -- Kiểm tra xem policy đã tồn tại hay chưa
    SELECT COUNT(*)
    INTO v_policy_exists
    FROM DBA_POLICIES
    WHERE object_owner = 'ATBM_ADMIN'
      AND object_name = 'THUONG'
      AND policy_name = 'THUONG_POLICY';

    -- Nếu policy đã tồn tại, xóa nó đi trước khi tạo lại
    IF v_policy_exists > 0 THEN
        EXECUTE IMMEDIATE 'BEGIN DBMS_RLS.DROP_POLICY(
            object_schema   => ''ATBM_ADMIN'',
            object_name     => ''THUONG'',
            policy_name     => ''THUONG_policy''
        ); END;';
    END IF;

    -- Tạo policy mới
    DBMS_RLS.ADD_POLICY(
        object_schema   => 'ATBM_ADMIN',
        object_name     => 'THUONG',
        policy_name     => 'THUONG_policy',
        policy_function => 'keys_access_predicate',
        statement_types => 'SELECT',
        update_check    => FALSE,
        enable          => TRUE
    );
END;
/
```

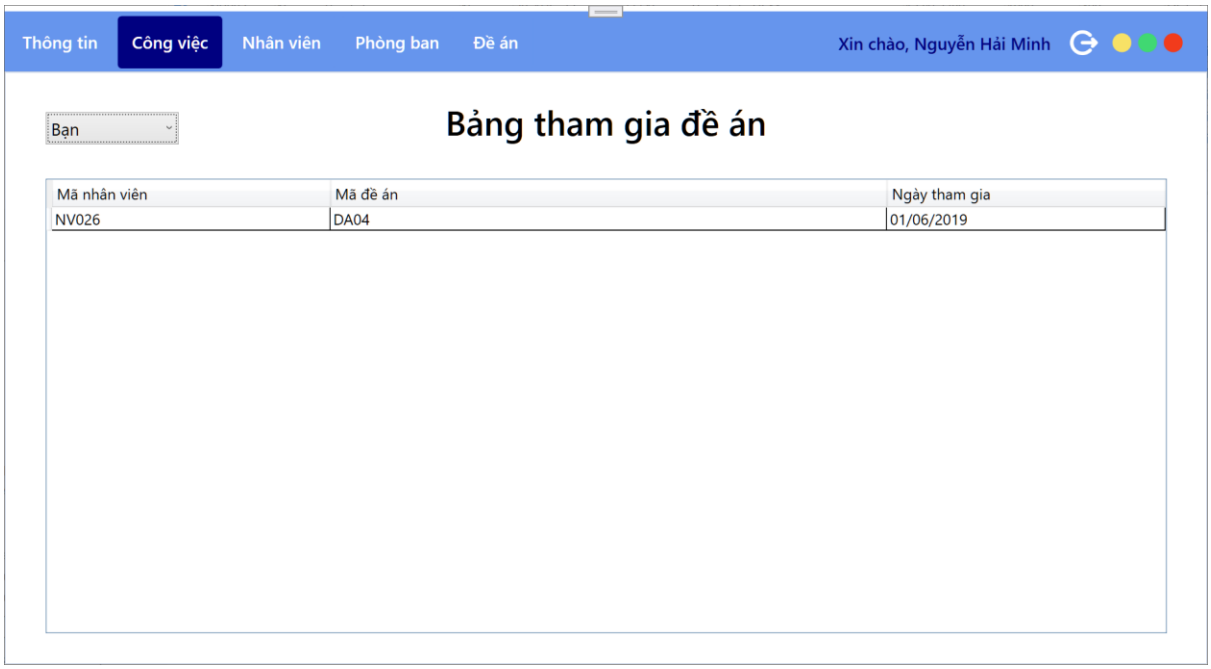
Tạo procedure cập nhật khóa cho Tài Chính và gán các quyền cần thiết cho nó.

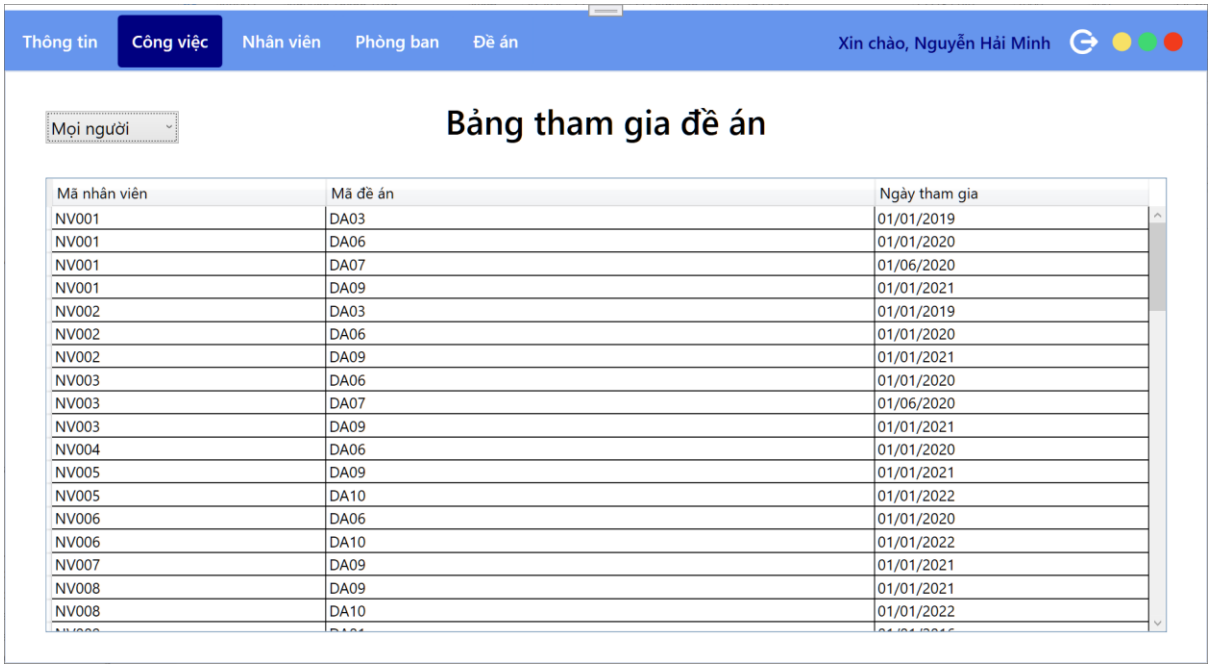
```
CREATE OR REPLACE PROCEDURE manage_THUONG (
    p_MANV IN THUONG.MANV%TYPE,
    p_DIPTHUONG IN THUONG.DIPTHUONG%TYPE,
    p_TIENTHUONG IN THUONG.TIENTHUONG%TYPE
)
IS
    v_count NUMBER;
BEGIN
    -- Kiểm tra xem MANV đã tồn tại trong bảng hay chưa
    SELECT COUNT(*)
    INTO v_count
    FROM THUONG
    WHERE MANV = p_MANV;

    IF v_count > 0 THEN
        -- Nếu MANV đã tồn tại, thực hiện cập nhật dữ liệu
        UPDATE THUONG
        SET DIPTHUONG = p_DIPTHUONG,
            TIENTHUONG = p_TIENTHUONG
        WHERE MANV = p_MANV;


    ELSE
        -- Nếu MANV chưa tồn tại, thực hiện chèn dữ liệu mới
        INSERT INTO THUONG (MANV, DIPTHUONG, TIENTHUONG)
        VALUES (p_MANV, p_DIPTHUONG, p_TIENTHUONG);


    END IF;

    COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;

END;

GRANT EXECUTE ON ATBM_ADMIN.manage_THUONG TO TAICHINH;
/

GRANT SELECT ON THUONG TO NHANVIEN;
/
GRANT INSERT, UPDATE ON THUONG TO TAICHINH;
/
```

Mức giao diện:

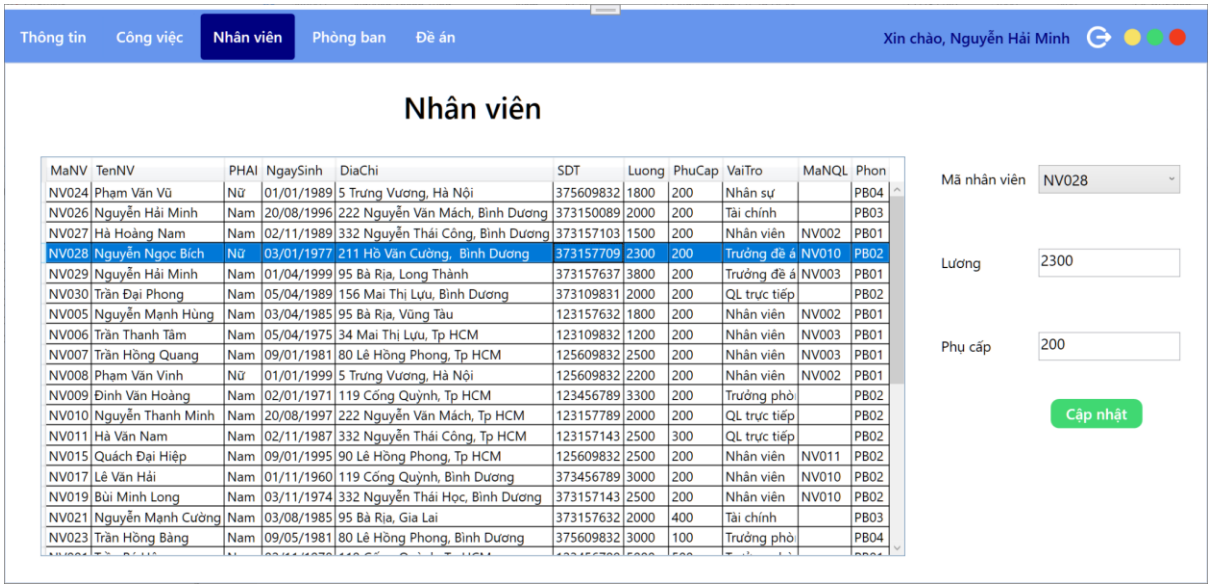
Ngoài các giao diện giống như một "Nhân viên", "Tài chính" còn có các màn hình giao diện khác.

Xem toàn bộ quan hệ PHANCONG:





Có thể xem và cập nhật toàn bộ lương của các nhân viên khác:



MaNV	TenNV	PHAI	NgaySinh	DiaChi	SDT	Luong	PhuCap	VaiTro	MaNQL	Phon
NV024	Phạm Văn Vũ	Nữ	01/01/1989	5 Trưng Vương, Hà Nội	375609832	1800	200	Nhân sự		PB04
NV026	Nguyễn Hải Minh	Nam	20/08/1996	222 Nguyễn Văn Mách, Bình Dương	373150089	2000	200	Tài chính		PB03
NV027	Hà Hoàng Nam	Nam	02/11/1989	332 Nguyễn Thái Công, Bình Dương	373157103	1500	200	Nhân viên	NV002	PB01
NV028	Nguyễn Ngọc Bích	Nữ	03/01/1977	211 Hồ Văn Cường, Bình Dương	373157709	2300	200	Trưởng đề á	NV010	PB02
NV029	Nguyễn Hải Minh	Nam	01/04/1999	95 Bà Rịa, Long Thành	373157637	3800	200	Trưởng đề á	NV003	PB01
NV030	Trần Đại Phong	Nam	05/04/1989	156 Mai Thị Lựu, Bình Dương	373109831	2000	200	QL trực tiếp		PB02
NV005	Nguyễn Mạnh Hùng	Nam	03/04/1985	95 Bà Rịa, Vũng Tàu	123157632	1800	200	Nhân viên	NV002	PB01
NV006	Trần Thanh Tâm	Nam	05/04/1975	34 Mai Thị Lựu, Tp HCM	123109832	1200	200	Nhân viên	NV003	PB01
NV007	Trần Hồng Quang	Nam	09/01/1981	80 Lê Hồng Phong, Tp HCM	125609832	2500	200	Nhân viên	NV003	PB01
NV008	Phạm Văn Vinh	Nữ	01/01/1999	5 Trưng Vương, Hà Nội	125609832	2200	200	Nhân viên	NV002	PB01
NV009	Đinh Văn Hoàng	Nam	02/01/1971	119 Cống Quỳnh, Tp HCM	123456789	3300	200	Trưởng phò		PB02
NV010	Nguyễn Thanh Minh	Nam	20/08/1997	222 Nguyễn Văn Mách, Tp HCM	123157789	2000	200	QL trực tiếp		PB02
NV011	Hà Văn Nam	Nam	02/11/1987	332 Nguyễn Thái Công, Tp HCM	123157143	2500	300	QL trực tiếp		PB02
NV015	Quách Đại Hiệp	Nam	09/01/1995	90 Lê Hồng Phong, Tp HCM	125609832	2500	200	Nhân viên	NV011	PB02
NV017	Lê Văn Hải	Nam	01/11/1960	119 Cống Quỳnh, Bình Dương	373456789	3000	200	Nhân viên	NV010	PB02
NV019	Bùi Minh Long	Nam	03/11/1974	332 Nguyễn Thái Học, Bình Dương	373157143	2500	200	Nhân viên	NV010	PB02
NV021	Nguyễn Mạnh Cường	Nam	03/08/1985	95 Bà Rịa, Gia Lai	373157632	2000	400	Tài chính		PB03
NV023	Trần Hồng Bàng	Nam	09/05/1981	80 Lê Hồng Phong, Bình Dương	375609832	3000	100	Trưởng phò		PB04

2.5. Chính sách 5: Nhân sự

Yêu cầu:

- Được quyền thêm, cập nhật trên quan hệ PHONGBAN.
- Thêm, cập nhật dữ liệu trong quan hệ NHANVIEN với giá trị các trường LUONG, PHUCAP là mang giá trị mặc định là NULL, không được xem LUONG, PHUCAP của người khác và không được cập nhật trên các trường LUONG, PHUCAP.

Ý tưởng: dùng procedure thêm và cập nhật trên quan hệ PHONGBAN cho role, dùng view với decode để bảo mật thông tin cột lương và phụ cấp, dùng procedure để bảo vệ cột lương, phụ cấp khi insert, update.

Chính sách	Đối tượng	Quyền
- Xem tất cả thuộc tính trên quan hệ NHANVIEN, không xem được LUONG, PHUCAP của người khác	(VIEW) NV_XemNHANVIEN	SELECT
- Thêm trên bảng PHONGBAN	(PROCEDURE) NS_THEM_PHONGBAN	EXECUTE

- Sửa trên bảng PHONGBAN	(PROCEDURE) NS_SUA_PHONGBAN	EXECUTE
- Thêm trên bảng NHANVIEN trừ cột LUONG, PHUCAP	(PROCEDURE) NS_THEM_NHANVIEN	EXECUTE
- Sửa trên bảng NHANVIEN trừ cột LUONG, PHUCAP	(PROCEDURE) NS_SUA_NHANVIEN	EXECUTE

Mức cơ sở dữ liệu:

```
--Được quyền thêm, cập nhật trên quan hệ PHONGBAN.
grant select on PHONGBAN to NHANSU;
/
CREATE OR REPLACE PROCEDURE NS_THEM_PHONGBAN (
    P_MAPHG     IN PHONGBAN.MAPHG%TYPE,
    P_TENPHG    IN PHONGBAN.TENPHG%TYPE,
    P_TRPHG     IN PHONGBAN.TRPHG%TYPE
) AS
BEGIN
    -- Thêm mới PHONGBAN
    INSERT INTO PHONGBAN (MAPHG, TENPHG, TRPHG)
    VALUES (P_MAPHG, P_TENPHG, P_TRPHG);
END;
/
CREATE OR REPLACE PROCEDURE NS_SUA_PHONGBAN (
    P_MAPHG     IN PHONGBAN.MAPHG%TYPE,
    P_TENPHG    IN PHONGBAN.TENPHG%TYPE,
    P_TRPHG     IN PHONGBAN.TRPHG%TYPE
) AS
BEGIN
    -- Cập nhật PHONGBAN
    UPDATE PHONGBAN
    SET TENPHG = P_TENPHG,
        TRPHG = P_TRPHG
    WHERE MAPHG = P_MAPHG;
END;
/
```

```
create or replace view NS_XEMNHANVIEN
as
    select  MANV,TENNV,PHAI,NGAYSINH,DIACHI,
    SODT, DECODE(MANV,sys_context('USERENV', 'CURRENT_USER'),luong,NULL) LUONG ,
    DECODE(MANV,sys_context('USERENV', 'CURRENT_USER'),PHUCAP,NULL) PHUCAP,
    VAITRO ,MANQL ,PHG
    from nhanvien
/
grant SELECT on NS_XEMNHANVIEN to NHANSU;
```

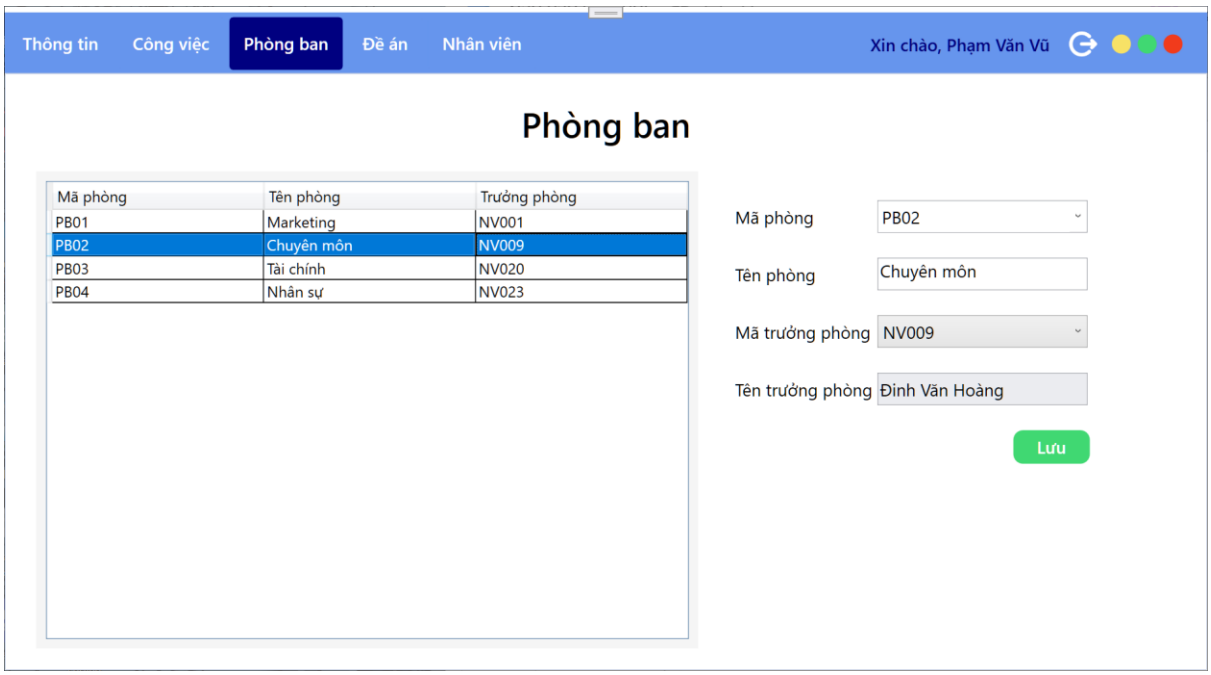
```
CREATE OR REPLACE PROCEDURE NS_THEM_NHANVIEN (
    P_MANV      IN NHANVIEN.MANV%TYPE,
    P_TENNV     IN NHANVIEN.TENNV%TYPE,
    P_PHAI      IN NHANVIEN.PHAI%TYPE,
    P_NGAYSINH  IN NHANVIEN.NGAYSINH%TYPE,
    P_DIACHI    IN NHANVIEN.DIACHI%TYPE,
    P_SODT      IN NHANVIEN.SODT%TYPE,
    P_VAITRO    IN NHANVIEN.VAITRO%TYPE,
    P_MANQL     IN NHANVIEN.MANQL%TYPE,
    P_PHG       IN NHANVIEN.PHG%TYPE
) AS
BEGIN
    -- Thêm mới NHANVIEN
    INSERT INTO NHANVIEN (MANV, TENNV, PHAI, NGAYSINH, DIACHI, SODT, LUONG, PHUCAP, VAITRO, MANQL, PHG)
    VALUES (P_MANV, P_TENNV, P_PHAI, P_NGAYSINH, P_DIACHI, P_SODT, NULL, NULL, P_VAITRO, P_MANQL, P_PHG);
    EXECUTE IMMEDIATE 'ALTER SESSION SET "_ORACLE_SCRIPT" = TRUE';
       execute immediate('CREATE USER ' || P_MANV || ' IDENTIFIED BY 1 DEFAULT TABLESPACE DA_ATBM');
      execute immediate('GRANT CREATE SESSION TO '|| P_MANV);
      execute immediate('GRANT NHANVIEN TO ' || P_MANV);
      IF P_VAITRO = 'Trưởng phòng' THEN
         execute immediate('GRANT TRUONGPHONG TO ' || P_MANV);
      ELSIF P_VAITRO = 'QL trực tiếp' THEN
         execute immediate('GRANT QLTRUCTIEP TO ' || P_MANV);
      ELSIF P_VAITRO = 'Tài chính' THEN
         execute immediate('GRANT TAICHINH TO ' || P_MANV);
      ELSIF P_VAITRO = 'Nhân sự' THEN
         execute immediate('GRANT NHANSU TO ' || P_MANV);
      ELSIF P_VAITRO = 'Trưởng đề án' THEN
         execute immediate('GRANT TRUONGDEAN TO ' || P_MANV);
      END IF;
      COMMIT;
END;
```

```
CREATE OR REPLACE PROCEDURE NS_SUA_NHANVIEN (
    P_MANV      IN NHANVIEN.MANV%TYPE,
    P_TENNV     IN NHANVIEN.TENNV%TYPE,
    P_PHAI      IN NHANVIEN.PHAI%TYPE,
    P_NGAYSINH  IN NHANVIEN.NGAYSINH%TYPE,
    P_DIACHI    IN NHANVIEN.DIACHI%TYPE,
    P_SODT      IN NHANVIEN.SODT%TYPE,
    P_VAITRO    IN NHANVIEN.VAITRO%TYPE,
    P_MANQL     IN NHANVIEN.MANQL%TYPE,
    P_PHG       IN NHANVIEN.PHG%TYPE
) AS
BEGIN
    -- Kiểm tra quyền của người dùng
    -- Ở đây bạn có thể thêm các điều kiện phù hợp để kiểm tra
    --quyền của người dùng trước khi cho phép cập nhật

    -- Cập nhật NHANVIEN
    UPDATE NHANVIEN
    SET TENNV = P_TENNV,
        PHAI = P_PHAI,
        NGAYSINH = P_NGAYSINH,
        DIACHI = P_DIACHI,
        SODT = P_SODT,
        VAITRO = P_VAITRO,
        MANQL = P_MANQL,
        PHG = P_PHG
    WHERE MANV = P_MANV;
END;
```

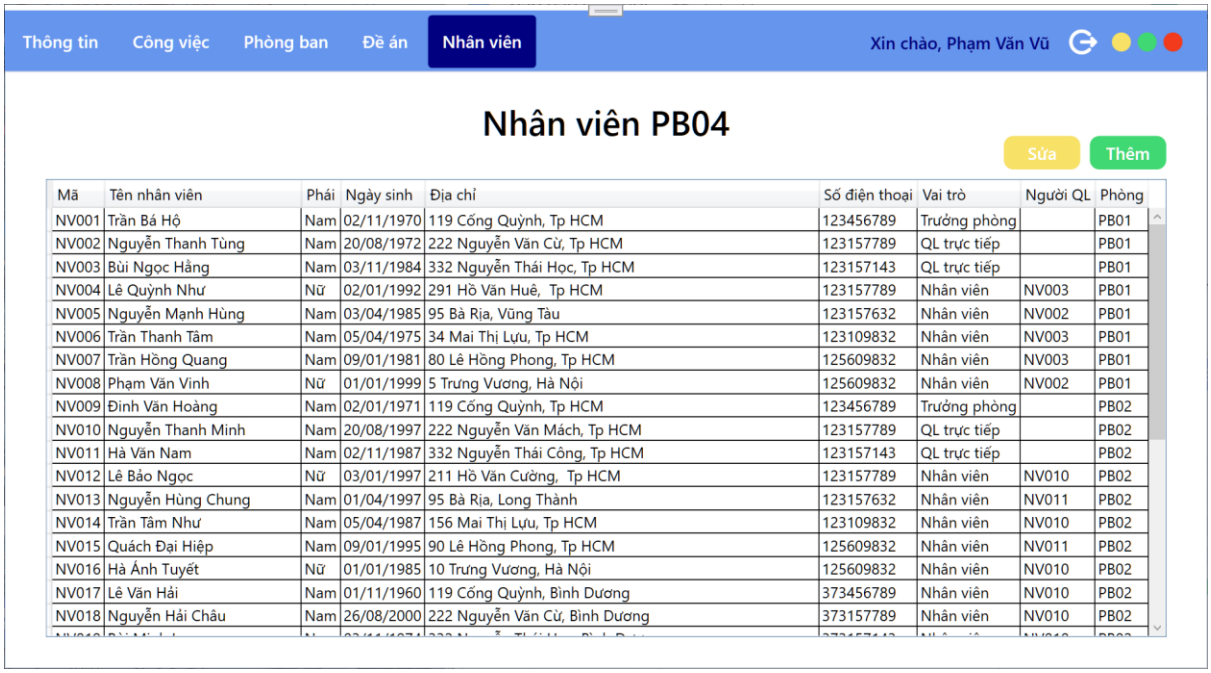
Mức giao diện:

Ngoài các giao diện giống như một “Nhân viên”, “Nhân sự” còn có các màn hình giao diện khác.

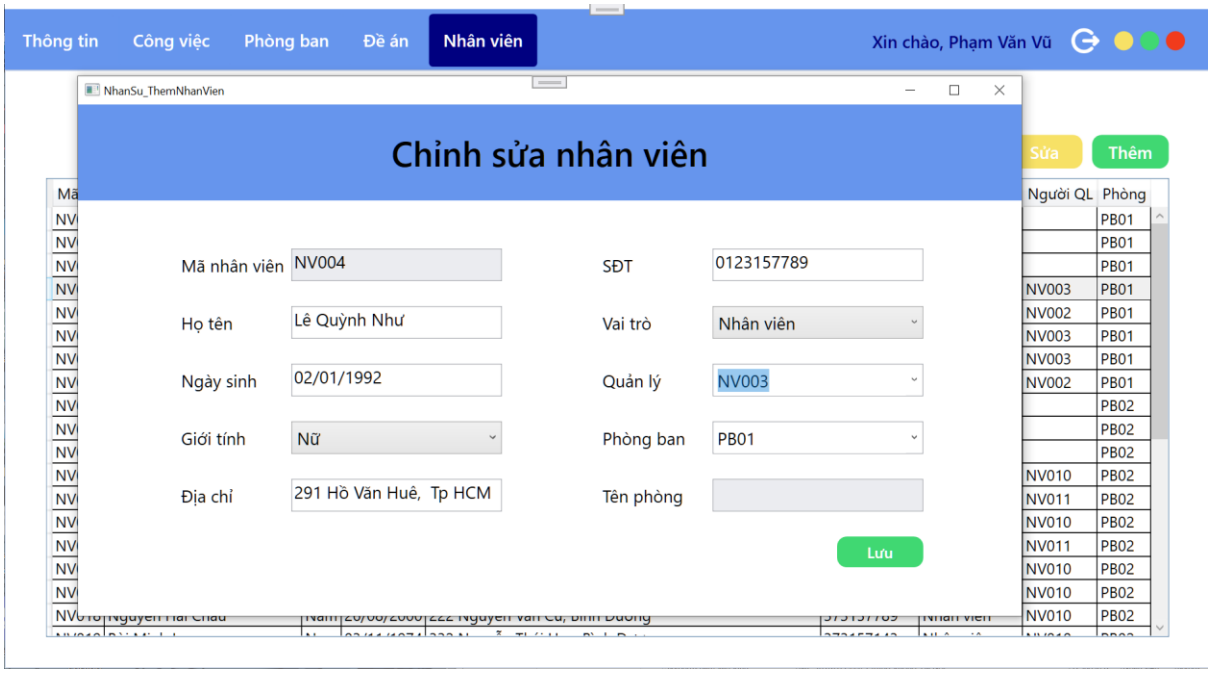
“Nhân sự” thêm, cập nhật trên bảng phân công:



Xem thông tin (trừ lương, phụ cấp) của toàn bộ nhân viên:



Thêm, sửa thông tin một nhân viên (trừ lương, phụ cấp):



2.6. Chính sách 6: Trưởng đề án

Ý tưởng:

Sử dụng BRAC. Tạo ra role TRUONGDEAN, sau đó tạo ra các view, proceduce mà đảm bảo các chính sách bảo mật của một TRUONGDEAN. Tiếp đó grant role này đến tất cả các USER có vai trò là "Trưởng đề án". Và một người có vai trò là "Trưởng đề án" sẽ có mọi quyền của role NHANVIEN.

Chính sách	Đối tượng	Quyền
Thêm trên bảng DEAN	(PROCEDURE) TRGDA_THEMDEAN	EXECUTE
Sửa trên bảng DEAN	(PROCEDURE) TRGDA_UPDATEDA	EXECUTE
Xóa DEAN	(PROCEDURE) TRGDA_XOADA	EXECUTE

Khi Xóa DEAN cần kiểm tra DEAN đó đã được phân công hay chưa (là khóa ngoại của PHANCONG)	(VIEW) TRGDA_XEMPHANCONG	SELECT

Mức cơ sở dữ liệu:

PROCEDURE: TRGDA_THEMDEAN

```
--Thêm đề án
CREATE OR REPLACE PROCEDURE TRGDA_THEMDEAN(
    MADA_ IN VARCHAR2,
    TENDA_ IN VARCHAR2,
    NGAYBD_ IN DATE,
    PHONG_ IN VARCHAR2
)
AS
BEGIN
    INSERT INTO NV_XemThongTinDeAn (MADA, TENDA, NGAYBD, PHONG)
    VALUES (MADA_, TENDA_, NGAYBD_, PHONG_);
    COMMIT;
END;
/
```

PROCEDURE: TRGDA_UPDATEDA

```
--Sửa đề án
CREATE OR REPLACE PROCEDURE TRGDA_UPDATEDA(
MADA_ IN VARCHAR2,
TENDA_ IN VARCHAR2,
NGAYBD_ IN DATE,
PHONG_ IN VARCHAR2)
AS
BEGIN
    UPDATE NV_XemThongTinDeAn
    SET TENDA = TENDA_, NGAYBD = NGAYBD_, PHONG = PHONG_
    WHERE MADA = MADA_;
    COMMIT;
END;
/
```

PROCEDURE: TRGDA_XOADA

```
--Xóa đề án
CREATE OR REPLACE PROCEDURE TRGDA_XOADA(MADA_ IN VARCHAR2)
AS
BEGIN
    DELETE FROM NV_XemThongTinDeAn WHERE MADA = MADA_;
    COMMIT;
END;
/
```

VIEW: TRGDA_XEMPHANCONG

```
CREATE OR REPLACE VIEW TRGDA_XEMPHANCONG
AS
    SELECT distinct MADA FROM PHANCONG;
```

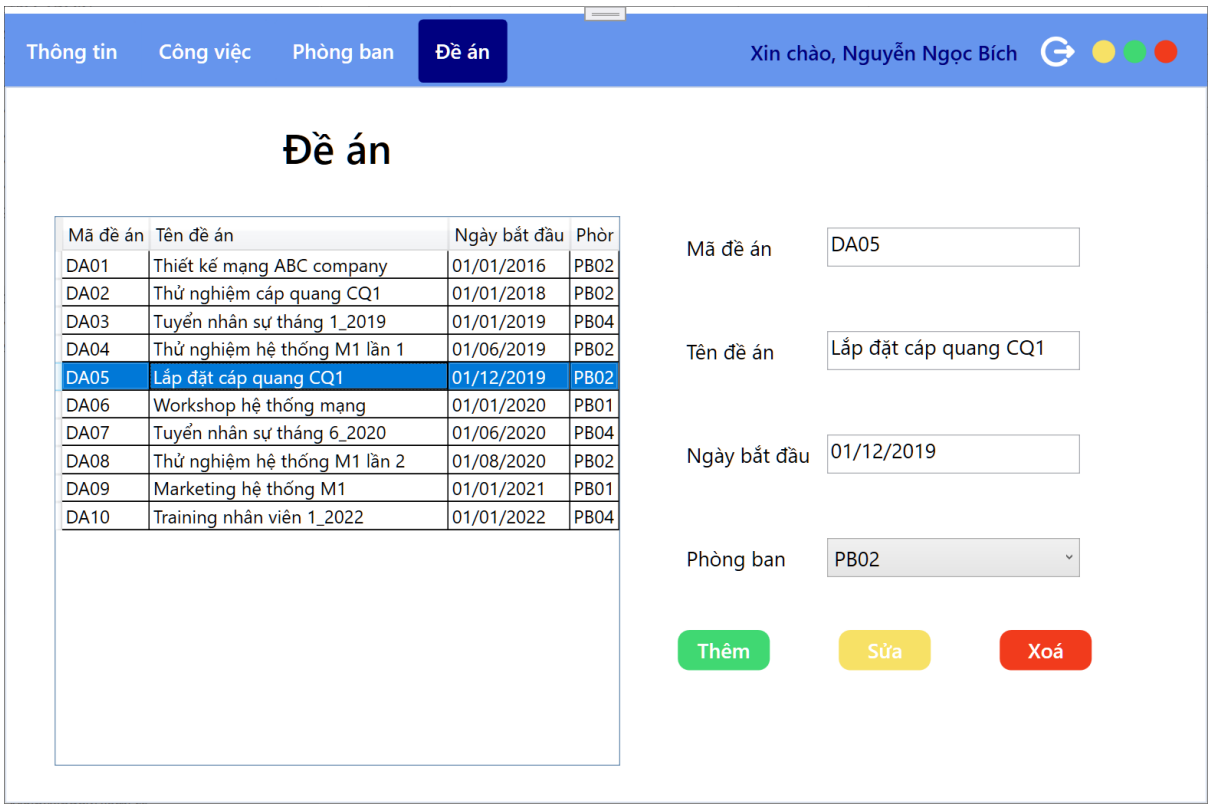
GRANT:

```
GRANT SELECT ON TRGDA_XEMPHANCONG TO TRUONGDEAN;
GRANT EXECUTE ON TRGDA_THEMDEAN TO TRUONGDEAN;
GRANT EXECUTE ON TRGDA_UPDATEDA TO TRUONGDEAN;
GRANT EXECUTE ON TRGDA_XOADA TO TRUONGDEAN;
/
```

Mức giao diện:

Ngoài các giao diện giống như một “Nhân viên”, “Trưởng đề án” còn có các màn hình giao diện khác.

Thêm, xóa, sửa trên bảng DEAN:



3. Mã hóa (chỉ cần cài đặt 1 chính sách, trình bày rõ PP quản lý khóa, trao đổi khóa, thay đổi khóa) và giao diện.

3.1. User vai trò nào sẽ thực hiện mã hóa?

Đối với các user có vai trò **nhân viên**, tất cả các user sẽ chỉ có quyền thực hiện **giải mã** lương và phụ cấp tương ứng của chính mình hiển thị trên màn hình thông tin cá nhân. **Không** có quyền thực hiện **giải mã** với lương và phụ cấp của các nhân viên **khác** và **không** được phép **mã hóa** bất kì thuộc tính nào.

Đối với các user có vai trò **tài chính,** tất cả các user sẽ có quyền thực hiện **giải mã và mã hóa** lương và phụ cấp của toàn bộ nhân viên trong hệ thống.

3.2. Mã hóa dữ liệu ở mức nào (mức cơ sở dữ liệu, mức ứng dụng, …)? Vì sao chọn mức mã hóa đề nghị?

- Mã hóa dữ liệu ở mức **ứng dụng**.Vì các lý do dưới:
- Mã hóa ở mức ứng dụng đảm bảo rằng dữ liệu đã được mã hóa trước khi lưu trữ trong cơ sở dữ liệu. Điều này đồng nghĩa rằng cơ sở dữ liệu không cần biết về việc

mã hóa và không thể truy cập vào dữ liệu gốc. Điều này tạo ra một lớp bảo mật bổ sung, ngay cả khi cơ sở dữ liệu bị tấn công.

- Mã hóa ở mức ứng dụng cho phép quản lý khóa một cách linh hoạt và độc lập. Có thể sử dụng các thuật toán mã hóa mạnh mẽ, quản lý và thay đổi khóa theo các phương thức tốt nhất. Điều này giúp bảo vệ dữ liệu và giảm nguy cơ việc mất khóa bí mật.
- Mã hóa ở mức ứng dụng cho phép kiểm soát chính xác quá trình mã hóa và giải mã dữ liệu. Có thể chọn mã hóa chỉ cho các thuộc tính cụ thể của ứng dụng hoặc dữ liệu quan trọng, giúp tối ưu hóa hiệu suất và quản lý tài nguyên.

3.3. Có cần thay đổi gì về cấu trúc lưu trữ dữ liệu hay không?

- **Kiểu dữ liệu**: Phải sử dụng các kiểu dữ liệu hỗ trợ mã hóa trong cơ sở dữ liệu của mình, chẳng hạn như kiểu dữ liệu BLOB hoặc VARCHAR2 để lưu trữ dữ liệu đã được mã hóa. Trong cơ sở dữ liệu, chúng em lựa chọn VARCHAR2 để linh hoạt hơn. Điều này đảm bảo rằng không có thông tin rõ ràng nào được lưu trữ trong cơ sở dữ liệu.
- **Độ dài trường dữ liệu**: Khi mã hóa dữ liệu, kích thước của dữ liệu đã mã hóa lớn hơn rất nhiều so với dữ liệu gốc. Vì vậy, cần xem xét lại độ dài của các trường dữ liệu trong cơ sở dữ liệu để đảm bảo rằng chúng đủ lớn để lưu trữ dữ liệu đã mã hóa.
- **Thay đổi quy trình truy xuất dữ liệu**: Khi truy xuất dữ liệu từ cơ sở dữ liệu, sẽ cần thực hiện quá trình giải mã dữ liệu đã được mã hóa. Điều này yêu cầu sự thay đổi trong quy trình truy xuất dữ liệu trong ứng dụng để đảm bảo rằng dữ liệu được giải mã và sử dụng một cách chính xác.
- **Quản lý khóa**: Khi mã hóa dữ liệu ở mức ứng dụng, sẽ cần quản lý khóa mã hóa để thực hiện quá trình mã hóa và giải mã. Điều này bao gồm việc lưu trữ khóa mã hóa và xác thực người dùng truy cập đúng khóa phù hợp.

3.4. Các khía cạnh của cơ chế quản lý khóa đề nghị: thiết lập khóa, lưu trữ khóa, phân phối khóa, phục hồi khóa khi người dùng quên khóa, thay khóa đồng loạt sau một thời gian.

3.4.1. Thiết lập khóa:

Hàm **GenerateAndSaveKeys** tạo cặp khóa **RSA** công khai và bí mật cho mỗi nhân viên dựa trên mã nhân viên và ngày tạo.

Cặp khóa được tạo bằng cách sử dụng lớp **RSACryptoServiceProvider** và lưu trữ dưới dạng chuỗi **Base64**.

```
private const int KeySize = 2048;//RSA 2048
public void GenerateAndSaveKeys(string employeeId)// hàm tạo cặp khóa key gồm container là manv+ngày tạo
{
    string publicKey;
    string privateKey;
    string containerName = $"KeyContainer_{DateTime.Now.ToString("yyyyMMdd")}_{employeeId}";


    CspParameters cspParams = new CspParameters
    {
        KeyContainerName = containerName
    };

    using (var rsa = new RSACryptoServiceProvider(KeySize, cspParams))
    {
        publicKey = Convert.ToBase64String(rsa.ExportCspBlob(false));
        privateKey = Convert.ToBase64String(rsa.ExportCspBlob(true));

        // Lưu khóa vào bảng trong Oracle
        SaveKeysToOracle(employeeId, publicKey, privateKey);
    }
}
```

3.4.2. Lưu trữ khóa:

Cặp khóa được lưu trữ trong cơ sở dữ liệu Oracle trong bảng **THUONG**.

Khóa công khai và khóa bí mật được lưu dưới dạng chuỗi **Base64**.

```
public void SaveKeysToOracle(string employeeId, string publicKey, string privateKey)// hàm lưu trữ khóa vào bảng trong oracle
{

    string procedureName = "ATBM_ADMIN.THUONG";
    OracleCommand command = new OracleCommand(procedureName, DB_Config.Conn);
    command.CommandType = CommandType.StoredProcedure;

    command.Parameters.Add(new OracleParameter("p_MANV", OracleDbType.Varchar2)).Value = employeeId;
    command.Parameters.Add(new OracleParameter("p_DIPTHUONG", OracleDbType.Varchar2)).Value = publicKey;
    command.Parameters.Add(new OracleParameter("p_TIENTHUONG", OracleDbType.Varchar2)).Value = privateKey;

    try
    {
        command.ExecuteNonQuery();

    }
    catch
    {

    }


}
```

3.4.3. Phân phối khóa:

Khóa công khai và khóa bí mật được lưu trữ trong cơ sở dữ liệu Oracle và có thể được truy xuất thông qua hàm **LoadPublicKeyFromOracle** và **LoadPrivateKeyFromOracle**.

```
public string LoadPublicKeyFromOracle(string employeeId)//hàm lấy ra publickey
{
    string publicKey = string.Empty;
    string query = "SELECT DIPTHUONG FROM ATBM_ADMIN.THUONG WHERE MANV = :employeeId";
    OracleCommand command = new OracleCommand(query, DB_Config.Conn);
    command.Parameters.Add(new OracleParameter("employeeId", OracleDbType.Varchar2)).Value = employeeId;

    try
    {
        object result = command.ExecuteScalar();
        publicKey = result != null ? result.ToString() : string.Empty;
    }
    catch (Exception ex)
    {
        Console.WriteLine("Error occurred: " + ex.Message);
    }

    return publicKey;
}
public string LoadPrivateKeyFromOracle(string employeeId)//hàm lấy ra privatekey
{
    string privateKey = string.Empty;
    string query = "SELECT TIENTHUONG FROM ATBM_ADMIN.THUONG WHERE MANV = :employeeId";
    OracleCommand command = new OracleCommand(query, DB_Config.Conn);
    command.Parameters.Add(new OracleParameter("employeeId", OracleDbType.Varchar2)).Value = employeeId;

    try
    {
        object result = command.ExecuteScalar();
        privateKey = result != null ? result.ToString() : string.Empty;
    }
    catch (Exception ex)
    {
        Console.WriteLine("Error occurred: " + ex.Message);
    }
    return privateKey;
}
```

4. OLS

Dựa theo đề bài chia thành:

3 level: giám đốc > trưởng phòng > nhân viên

3 compartment: mua bán, sản xuất, gia công

3 group: miền Bắc, miền Trung, miền Nam

a. Hãy gán nhãn cho 03 người dùng trong hệ thống:

01 giám đốc có thể đọc được toàn bộ dữ liệu

Giám đốc

Giám đốc: mua bán, sản xuất, gia công: miền Bắc, miền Trung, miền Nam

01 trưởng phòng phụ trách lĩnh vực sản xuất miền Nam

Trưởng phòng: sản xuất: miền Nam

01 giám đốc phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Bắc (có thể đọc được toàn bộ dữ liệu theo đúng cấp bậc và không phân biệt lĩnh vực).

Giám đốc: mua bán, sản xuất, gia công: miền Bắc

b. Hãy cho biết cách thức phát tán dòng thông báo t1 đến tất cả trưởng phòng phụ trách tất cả các lĩnh vực không phân biệt chi nhánh.

T1: Trưởng phòng: mua bán, sản xuất, gia công: miền Bắc, miền Trung, miền Nam

c. Hãy cho biết cách thức phát tán dòng thông báo t2 đến trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung.

T2: Trưởng phòng: sản xuất: miền Trung

d. Em hãy cho thêm một số kịch bản phát tán dữ liệu nữa trên mô hình OLS đã cài đặt.

Hãy cho biết cách thức phát tán dòng thông báo t3 đến Giám đốc phụ trách lĩnh vực mua bán ở cả nước.

T3: Giám đốc: mua bán: miền Bắc, miền Trung, miền Nam

Hãy cho biết cách thức phát tán dòng thông báo t4 đến Giám đốc phụ trách tất cả các lĩnh vực ở miền Nam.

T4: Giám đốc: mua bán, sản xuất, gia công: miền Nam

Hãy cho biết cách thức phát tán dòng thông báo t5 đến Nhân viên thuộc lĩnh vực gia công ở miền Trung.

T5: Nhân viên: gia công: miền Trung

Cài đặt: Bị lỗi ORA-65109 OPERATION NOT ALLOWED IN CBD$ROOT tại bước create OLS policy

5. Audit cơ bản và FGA (4 chính sách). (Hà)

Auditing dùng để ghi nhận lại những sự việc đã diễn ra và có thông báo thích hợp.

Việc audit có hiệu quả khi có kế hoạch vì nếu audit tất cả các hành động của mọi user trên cơ sở dữ liệu là một việc gây lãng phí tài nguyên, làm chậm hệ thống và với lượng dữ liệu khổng lồ khi ghi vết tất cả hành động sẽ khó khăn khi đọc lại để tìm ra vấn đề.

5.1. Audit cơ bản

Kiểm tra cơ bản cung cấp các công cụ để ghi lại các hoạt động của người dùng trên cơ sở dữ liệu Oracle

Application Server Log

Application Auditing

Trigger Auditing

5.2. Fine – grained Auditing (FGA)

Là một tính năng được Oracle hỗ trợ có thể audit:

- Trên đối tượng là table/ view.
- Việc thực thi procedure.
- Các đặc quyền hệ thống (VD: tắt kích hoạt 1 trigger).
- Trên 1 số user cụ thể.
- Trên các hành động thành công hoặc không thành công.
- kiểm tra điều kiện trước khi audit, column sensitivity, …
- Cú pháp:

```
DBMS_FGA.ADD_POLICY (
        object_schema   IN VARCHAR2,
        object_name  IN VARCHAR2,
        policy_name  IN VARCHAR2,
        statement_types IN VARCHAR2,
        audit_condition IN VARCHAR2,
        audit_columnIN VARCHAR2,
handler_schema VARCHAR2,
handler_module VARCHAR2,
```

enable BOOLEAN,

audit_trail BINARY_INTEGER IN DEFAULT,

audit_column_opts BINARY_INTEGER IN DEFAULT);

Tên	**Mô tả**	**Mặc định**
object_schema	Tên schema chứa đối tượng, nếu NULL sẽ lấy schema user hiện tại	NULL
object_name	Tên đối tượng	
policy_name	Tên chính sách kiểm tra	
statement_types	Loại câu lệnh (INSERT, UPDATE, DELETE, SELECT, ...)	SELECT
audit_condition	Điều kiện kiểm tra (tùy chọn), nếu NULL hàng nào cũng sẽ giám sát	NULL
audit_column	Tên cột kiểm tra (tùy chọn), nếu NULL cột nào cũng sẽ giám sát	NULL
handler_schema	Tên schema chứ hàm xử lí sự kiện, nếu NULL sẽ lấy schema user hiện tại	NULL
handler_module	Tên hàm xử lí sự kiện	NULL
enable	TRUE là chính sách được kích hoạt	FALSE
audit_trail	Nơi ghi lại bản giám sát	DB+EXTENDE
audit_column_opts	Chỉ ra giám sát tới một trong những cột trong audit_column hay tất cả các cột	ANY_COLUMNS

5.3. Các chính sách

a. Những người đã cập nhật trường THOIGIAN trong quan hệ PHANCONG.

Ý tưởng: Chỉ lưu vết nếu dữ liệu thời gian thay đổi. Sử dụng trigger thay vì FGA vì trigger dùng được dữ liệu :old và :new để so sánh.

Cài đặt:

Tạo bảng lưu dữ liệu cần auditing

```
CREATE TABLE AUD_PHANCONG
(
audit_id number, --tao 1 truong null lam khoa
USRNAME VARCHAR2(10),
ACTION VARCHAR2(6),
NV_ID VARCHAR2(5),
DA_ID VARCHAR2(5),
COLUMN_NAME VARCHAR2(255),
CLIENT_ID VARCHAR2(255),
OLD_VALUE VARCHAR2(10),
NEW_VALUE VARCHAR2(10),
ACTION_DATE timestamp
);
```

Tạo trigger update trên trường THOIGIAN bảng PHANCONG

```
CREATE OR REPLACE TRIGGER audit_thgian_pc
BEFORE UPDATE OF THOIGIAN ON PHANCONG
FOR EACH ROW
BEGIN
    -- Kiểm tra nếu trường THOIGIAN đã thay đổi
    IF :OLD.THOIGIAN <> :NEW.THOIGIAN THEN
        INSERT INTO AUD_phancong (audit_id, USRNAME, ACTION, NV_ID, DA_ID, COLUMN_NAME, CLIENT_ID, OLD_VALUE, NEW_VALUE, ACTION_DATE)
            VALUES (null, user,
            --sys_context('USERENV', 'CURRENT_USER'),
            'UPDATE',:OLD.MANV, :OLD.MADA,'THOIGIAN', NULL,
            --sys_context('USERENV', 'CLIENT_IDENTIFIED'),
            TO_CHAR(:OLD.THOIGIAN), TO_CHAR(:NEW.THOIGIAN), SYSDATE);
    END IF;
END;
/
```

Kết quả:

```
--SELECT * FROM AUD_PHANCONG;
```

Script Output | Query Result

SQL | All Rows Fetched: 3 in 0.003 seconds

	USRNAME	ACTION	COLUMN_N...	CLIENT_ID	OLD_VALUE	NEW_VALUE	ACTION_DATE	NV_ID	DA_ID	AUD...
1	NV009	UPDATE	THOIGIAN	(null)	01-JAN-16	01-JAN-23	29-JUN-23 04.15.22.000000000 PM	NV009	DA01	(n...
2	NV025	UPDATE	THOIGIAN	(null)	01-JAN-20	01-JAN-23	29-JUN-23 04.21.04.000000000 PM	NV002	DA06	(n...
3	NV025	UPDATE	THOIGIAN	(null)	01-JAN-23	01-JAN-20	29-JUN-23 04.21.08.000000000 PM	NV002	DA06	(n...

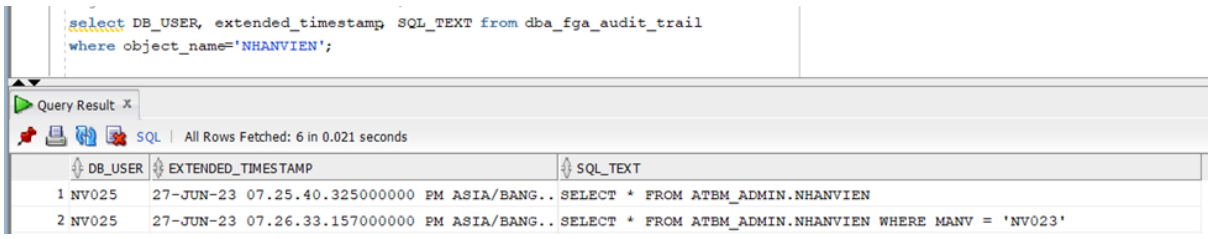
b. Những người đã đọc trên trường LUONG và PHUCAP của người khác.

Ý tưởng: Sử dụng FGA do Oracle cung cấp, ưu tiên xài FGA đỡ phải tự cài đặt, cập nhật nếu có.

Cài đặt: vị từ dùng sys_context current_user để kiểm tra người dùng hiện tại

```
BEGIN
    DBMS_FGA.ADD_POLICY(
        OBJECT_SCHEMA => 'ATBM_ADMIN',
        OBJECT_NAME => 'NHANVIEN',
        POLICY_NAME => 'SLT_LG_PC_NHANVIEN',
        AUDIT_COLUMN => 'LUONG, PHUCAP',
        AUDIT_CONDITION => 'MANV != sys_context(''USERENV'', ''CURRENT_USER'')',
        ENABLE => TRUE,
        STATEMENT_TYPES => 'SELECT'
        --AUDIT_COLUMN_OPTS => DBMS_FGA.ALL_COLUMNS
    );
END;
/
```

Kết quả:



```
select DB_USER, extended_timestamp, SQL_TEXT from dba_fga_audit_trail
where object_name='NHANVIEN';
```

Query Result | SQL | All Rows Fetched: 6 in 0.021 seconds

	DB_USER	EXTENDED_TIMESTAMP	SQL_TEXT
1	NV025	27-JUN-23 07.25.40.325000000 PM ASIA/BANG..	SELECT * FROM ATBM_ADMIN.NHANVIEN
2	NV025	27-JUN-23 07.26.33.157000000 PM ASIA/BANG..	SELECT * FROM ATBM_ADMIN.NHANVIEN WHERE MANV = 'NV023'

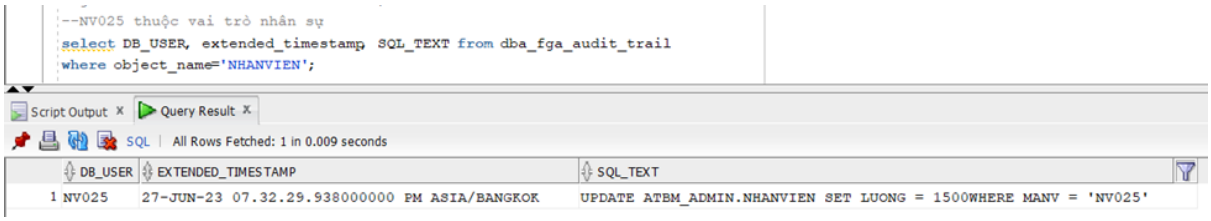
c. **Một người không thuộc vai trò "Tài chính" nhưng đã cập nhật thành công** trên **trường LUONG và PHUCAP.**

Ý tưởng: Sử dụng FGA do Oracle cung cấp, ưu tiên xài FGA đỡ phải tự cài đặt, cập nhật nếu có.

Cài đặt: vị từ VAITRO != "Tài chính"

```
BEGIN
    DBMS_FGA.ADD_POLICY(
        OBJECT_SCHEMA => 'ATBM_ADMIN',
        OBJECT_NAME => 'NHANVIEN',
        POLICY_NAME => 'UPD_LG_PC_NVTC',
        AUDIT_COLUMN => 'LUONG, PHUCAP',
        AUDIT_CONDITION => 'VAITRO != ''Tài chính''',
        ENABLE => TRUE,
        STATEMENT_TYPES => 'UPDATE'
    );
END;
```

Kết quả:



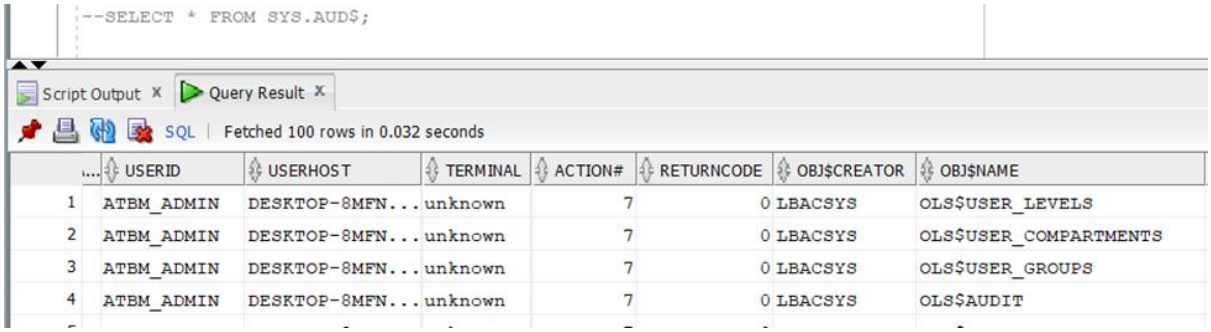
```
--NV025 thuộc vai trò nhân sự
select DB_USER, extended_timestamp, SQL_TEXT from dba_fga_audit_trail
where object_name='NHANVIEN';
```

Script Output | Query Result | SQL | All Rows Fetched: 1 in 0.009 seconds

	DB_USER	EXTENDED_TIMESTAMP	SQL_TEXT
1	NV025	27-JUN-23 07.32.29.938000000 PM ASIA/BANGKOK	UPDATE ATBM_ADMIN.NHANVIEN SET LUONG = 1500WHERE MANV = 'NV025'

d. Kiểm tra nhật ký hệ thống.

Nhật kí hệ thống được lưu trữ trong bảng SYS.AUD$



```
--SELECT * FROM SYS.AUD$;
```

Script Output | Query Result | SQL | Fetched 100 rows in 0.032 seconds

	USERID	USERHOST	TERMINAL	ACTION#	RETURNCODE	OBJ$CREATOR	OBJ$NAME
1	ATBM_ADMIN	DESKTOP-8MFN...	unknown	7	0	LBACSYS	OLS$USER_LEVELS
2	ATBM_ADMIN	DESKTOP-8MFN...	unknown	7	0	LBACSYS	OLS$USER_COMPARTMENTS
3	ATBM_ADMIN	DESKTOP-8MFN...	unknown	7	0	LBACSYS	OLS$USER_GROUPS
4	ATBM_ADMIN	DESKTOP-8MFN...	unknown	7	0	LBACSYS	OLS$AUDIT

TÀI LIỆU THAM KHẢO

https://docs.oracle.com/en/database/oracle/oracle-database/19/olsag/creating-an-oracle-label-security-policy.html#GUID-4B346D40-A65C-44F4-A4C3-A684F3436942

https://aithao0007.files.wordpress.com/2013/09/lab-06.pdf

https://tailieuebook.com/tai-lieu-thuc-hanh-bao-mat-he-thong-thong-tin-bai-thuc-hanh-so-13-fine-grained-auditing-3218/