

Prueba de Caja Blanca

“Detección de Blancos Biológicos en Rosales”

Versión 1.1

Integrantes:

Nicolas Cedillo
Alisson Clavijo
Lizzette Zapata

Fecha 2024-01-11

Ingreso del Usuario (LOGIN)

CÓDIGO FUENTE

```
@api_router.post("/login")
async def login(response: Response, form: OAuth2PasswordRequestForm = Depends()):
    """
    Endpoint para autenticar al usuario y generar un token de acceso.

    Args:
        response (Response): Objeto de respuesta HTTP.
        form (OAuth2PasswordRequestForm): Datos del formulario de inicio de
    sesión.

    Returns:
        Response: Objeto de respuesta HTTP con el token de acceso y tipo de
    token.

    Raises:
        HTTPException: Si la contraseña es incorrecta.
    """
    user: UserDB = search_user(form.username)
    print(form.username)

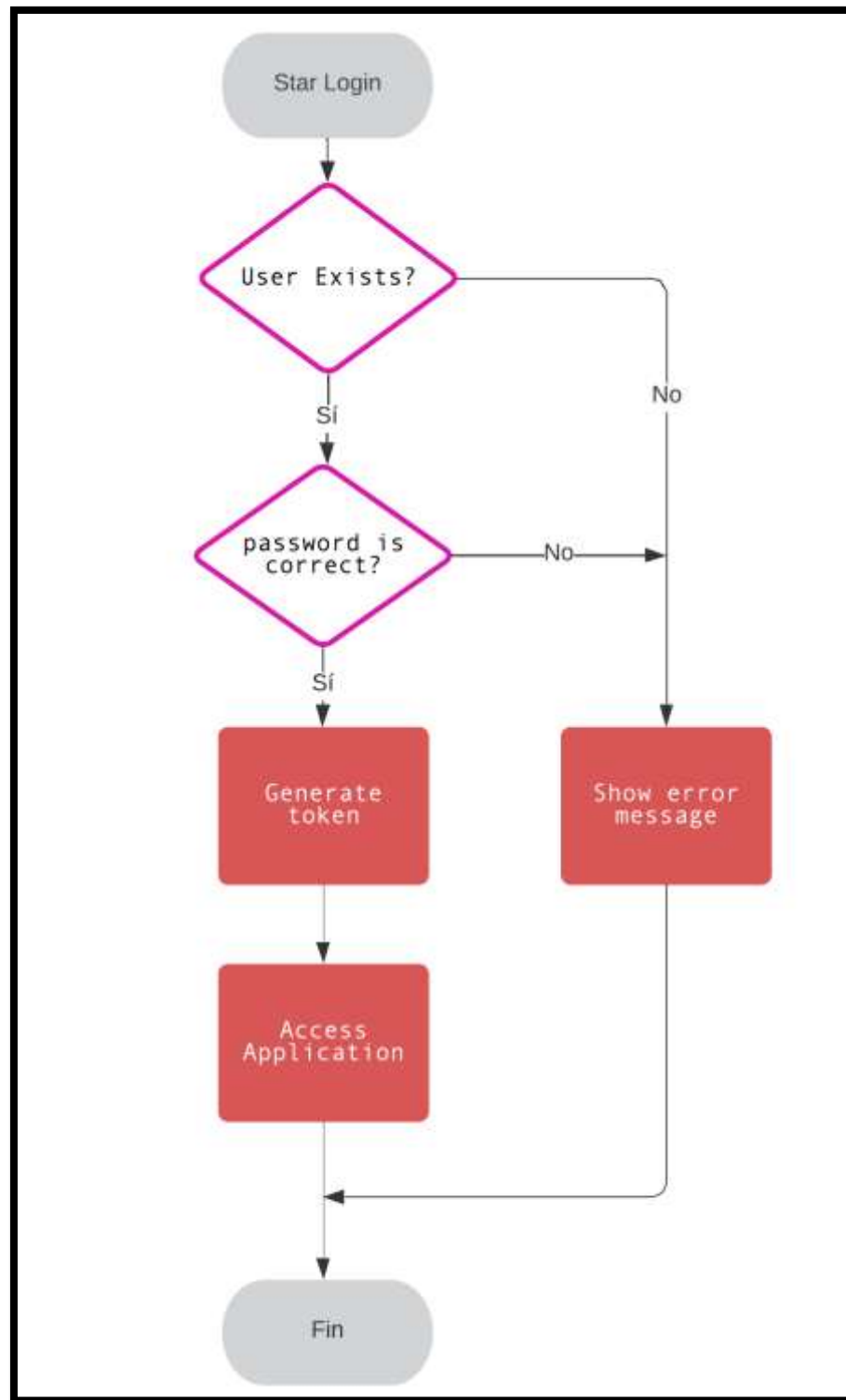
    if not crypt.verify(form.password, user.user_password):
        raise HTTPException(
            status_code=status.HTTP_401_UNAUTHORIZED, detail="Incorrect
password")

    expire = ((datetime.utcnow() +
timedelta(minutes=ACCESS_TOKEN_DURATION)).timestamp())

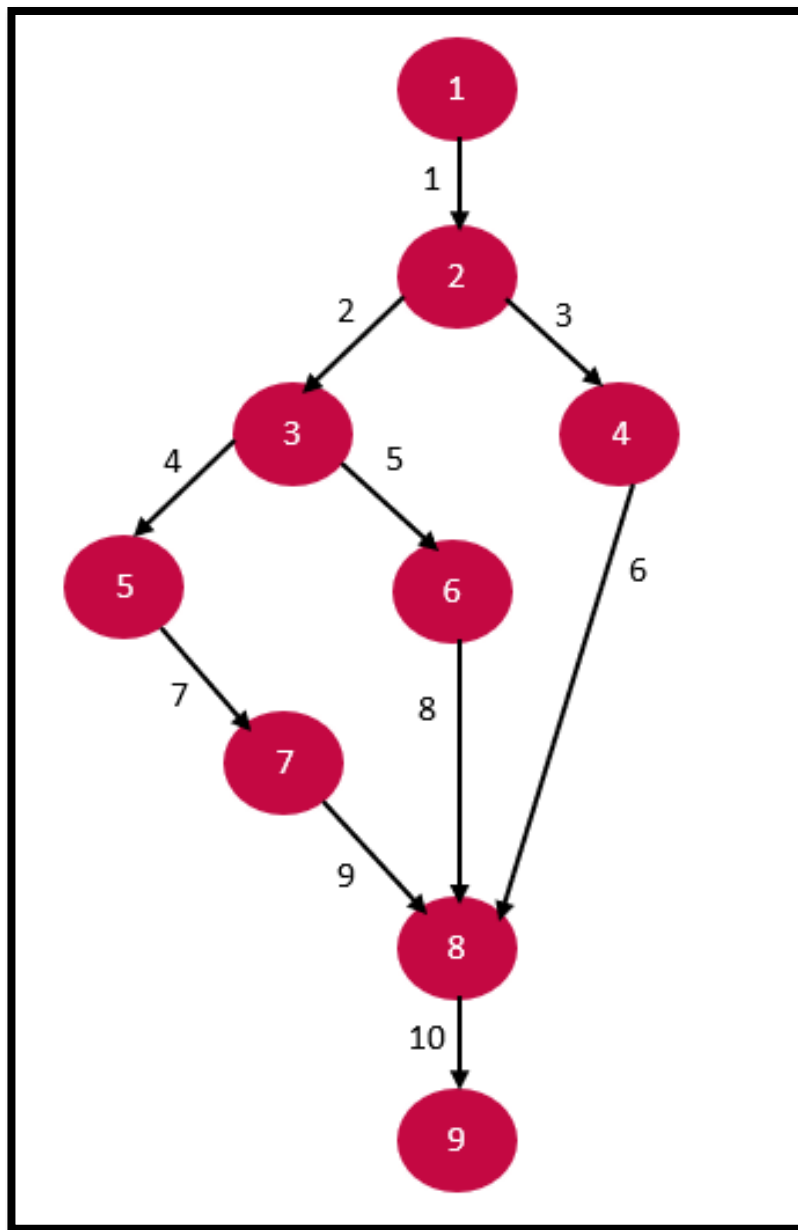
    access_token = {
        "sub": user.user_name,
        "exp": expire
    }

    encoded_token = jwt.encode(access_token, SECRET, algorithm=ALGORITHM)
    response = JSONResponse(
        content={"access_token": encoded_token, "token_type": "bearer"},
        status_code=status.HTTP_200_OK
    )
    response.set_cookie(
        key="access_token",
        value=encoded_token,
        expires=expire,
        httponly=True,
        samesite='lax',
    )
    return response
```

Diagrama de Flujo



GRAFO



RUTAS

R1: 1,2,4,7,9,10

R2: 1,2,5,8,10

R3: 1,2,3,6,10

COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

- $V(G) = A - N + 2 = 3$
- $V(G) = 10 - 9 + 2 = 3$

DONDE:

A: Número de aristas

N: Número de nodos