

ĐỒ ÁN MÔN HỌC

Kiểm Thử Hệ Thống Số Liên Lạc Điện Tử

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên Ngành: **Công Nghệ Phần Mềm**

Sinh viên thực hiện :

Đỗ Hoàng Đại	1711060081	17DTHA3
Nguyễn Tấn Thái	1711060694	17DTHA3
Vũ Hoàng Phong	1711060251	17DTHA3

Mục Lục

Mô Tả Kiểm Thử Chức Năng App Ứng Dụng Sổ Liên Lạc Điện Tử https://eschool-usteam.tk/	3
Kiểm thử hộp đen	4
Kiểm thử hộp trắng	9
Kiểm Thử Lướt Tải Với Công Cụ LoadStorm.....	21
Kiểm Thử Website với công cụ Selenium	25
Công Cụ Nessus	29

Mô Tả Kiểm Thử Chức Năng App Ứng Dụng Sổ Liên Lạc Điện Tử <https://eschool-usteam.tk/>

1) Đăng nhập (admin)

Đặc Tả Chức Năng: chức năng đăng nhập dành cho phía nhà trường, khi đăng nhập vào hệ thống nếu đúng mật khẩu hệ thống sẽ đẩy vào trang quản lý nếu nhập sai sẽ vào các trường hợp sau:

Không nhập tên đăng nhập: hệ thống thông báo nhập tên đăng nhập

Không nhập mật khẩu: hệ thống thông báo không nhập mật khẩu

Không nhập tên đăng nhập và mật khẩu: hệ thống thông báo nhập tên đăng nhập và mật khẩu

Nhập đúng tên sai mật khẩu: hệ thống thông báo sai mật khẩu

Nhập sai mật khẩu sai tên đăng nhập: hệ thống thông báo sai tên đăng nhập hoặc mật khẩu và cho phép người dùng nhập lại

Nhập đúng mật khẩu và tên đăng nhập: chuyển sang trang index hệ thống

2) Nhập điểm (giáo viên)

Đặc Tả Chức Năng: Khi GV đăng nhập vào hệ thống chọn chức năng nhập điểm, giáo viên sẽ nhập điểm vào hệ thống, hệ thống check điểm giáo viên nhập vào.

Thang điểm giáo viên nhập: $0 \leq x \leq 10$

3) Thời Khóa Biểu (admin)

Đặc Tả Chức Năng: khi admin đăng nhập vào hệ thống sẽ có chức năng quản lý thời khóa biểu, để quản lý thời khóa biểu từng lớp, hệ thống sẽ chia theo khối, lớp, thứ. Khi chọn theo thứ tự chọn khối, chọn lớp, chọn thứ.

4) Đăng Ký Bảo Hiểm Y Tế (user)

Đặc Tả Chức Năng: Admin đăng nhập vào hệ thống sẽ có quản lý được thời gian mở hoặc đóng cổng đăng ký bảo hiểm y tế cho phía phụ huynh, bao gồm ngày bắt đầu và kết thúc

Kiểm thử hộp đen

Bảng quyết định: Chức Năng Đăng Nhập

Các điều kiện	Tài khoản	B	B	B	I	I	I	V	V	V
	Mật khẩu	B	I	V	B	I	V	B	I	V
Các hành động	Thông báo lỗi	M1	M2	M2	M3	M4	M4	M3	M4	
	Chuyển đến trang	L	L	L	L	L	L	L	L	A

M1 : Hãy Nhập Tên Đăng Nhập ! Và Hãy Nhập Mật Khẩu !

M2: Hãy Nhập Tên Đăng Nhập !

M3: Hãy Nhập Mật Khẩu !

M4: Tên đăng nhập hoặc mật khẩu không đúng !

L: Trang login

A: Trang home admin

Tài khoản	Mật khẩu	Đầu ra thực tế	Đầu ra mong muốn
-	-	Hãy Nhập Tên Đăng Nhập ! và Hãy Nhập Mật Khẩu !	Hãy Nhập Tên Đăng Nhập ! Và Hãy Nhập Mật Khẩu !
-	abcde	Hãy Nhập Tên Đăng Nhập !	Hãy Nhập Tên Đăng Nhập !
-	admin	Hãy Nhập Tên Đăng Nhập !	Hãy Nhập Tên Đăng Nhập !
Abcde123	-	Hãy Nhập Mật Khẩu !	Hãy Nhập Mật Khẩu !
Abcde123	Abcde456	Tên đăng nhập hoặc mật khẩu không đúng !	Tên đăng nhập hoặc mật khẩu không đúng !
Abcde123	admin	Tên đăng nhập hoặc mật khẩu không đúng !	Tên đăng nhập hoặc mật khẩu không đúng !
admin	-	Hãy Nhập Mật Khẩu !	Hãy Nhập Mật Khẩu !
admin	Abcde456	Tên đăng nhập hoặc mật khẩu không đúng !	Tên đăng nhập hoặc mật khẩu không đúng !

admin	admin		
-------	-------	--	--

Bảng quyết định: Chức Năng Quản Lý Truy Cập Thời Khóa Biểu

Điều kiện	Chọn khối	B	B	B	B	V	V	V
	Chọn lớp	B	B	V	V	B	V	V
	Chọn thứ	B	V	B	V	B	B	V
hành động	Thông báo lỗi	M1	M1	M1	M1	M1	M1	
	Hiện thị thông tin	F	F	F	F	F	F	T

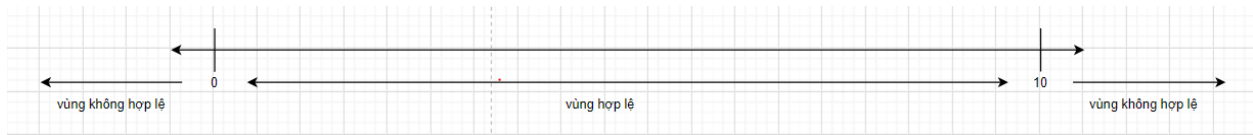
M1: vui lòng điền đầy đủ thông tin

F: Hiện thị thông báo lỗi

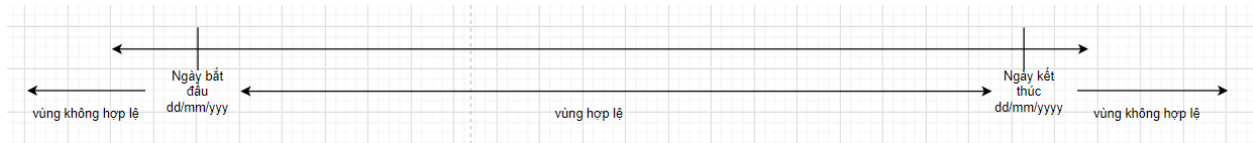
T: Hiện thị thông tin tìm được

Chọn khối	Chọn lớp	Chọn thứ	Đầu ra thực tế	Đầu ra mong muốn
-	-	-	vui lòng điền đầy đủ thông tin	vui lòng điền đầy đủ thông tin
-	-	2	vui lòng điền đầy đủ thông tin	vui lòng điền đầy đủ thông tin
-	11B3	-	vui lòng điền đầy đủ thông tin	vui lòng điền đầy đủ thông tin
-	11B3	2	vui lòng điền đầy đủ thông tin	vui lòng điền đầy đủ thông tin
10	-	-	vui lòng điền đầy đủ thông tin	vui lòng điền đầy đủ thông tin
10	10C2	-	vui lòng điền đầy đủ thông tin	vui lòng điền đầy đủ thông tin
10	10C2	3		

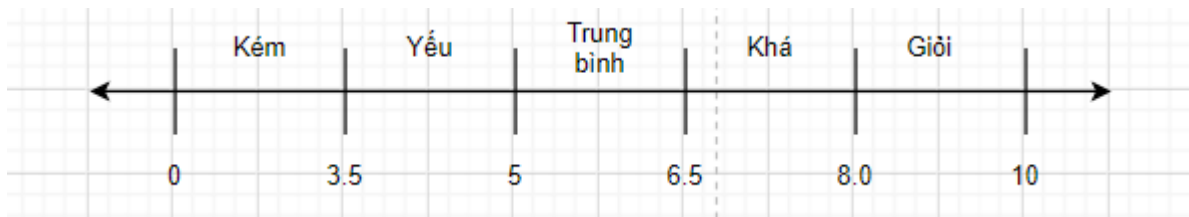
-Vùng tương đương: chức năng nhập điểm



-Vùng tương đương: chức năng đăng ký bảo hiểm y tế



-Vùng giá trị biên: chức năng xét học lực



+Đoạn “kém”: 0 và 3.4

+Đoạn “yếu”: 3.5 và 4.9

+Đoạn "trung bình": 5.0 và 6.4

+Đoạn “khá”: 6.5 và 7.9

+Đoạn “giỏi”: 8.0 và 10

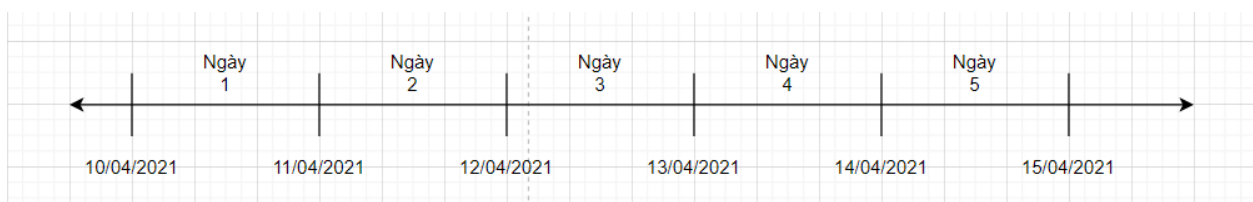
+Đoạn trước 0 (số âm) hoặc sau 10 (> 10) là vùng ngoại biên

-Vùng giá trị biên: chức năng đăng ký bảo hiểm y tế

Giả sử: Ngày đăng ký bảo hiểm y tế

+ Ngày bắt đầu: 10/04/2021

+ Ngày kết thúc: 15/04/2021



+Đoạn “Ngày 1”: 10/04/2021(0h1p) -> 11/04/2021(23h59p)

+Đoạn “Ngày 2”: 11/04/2021(0h1p) -> 12/04/2021(23h59p)

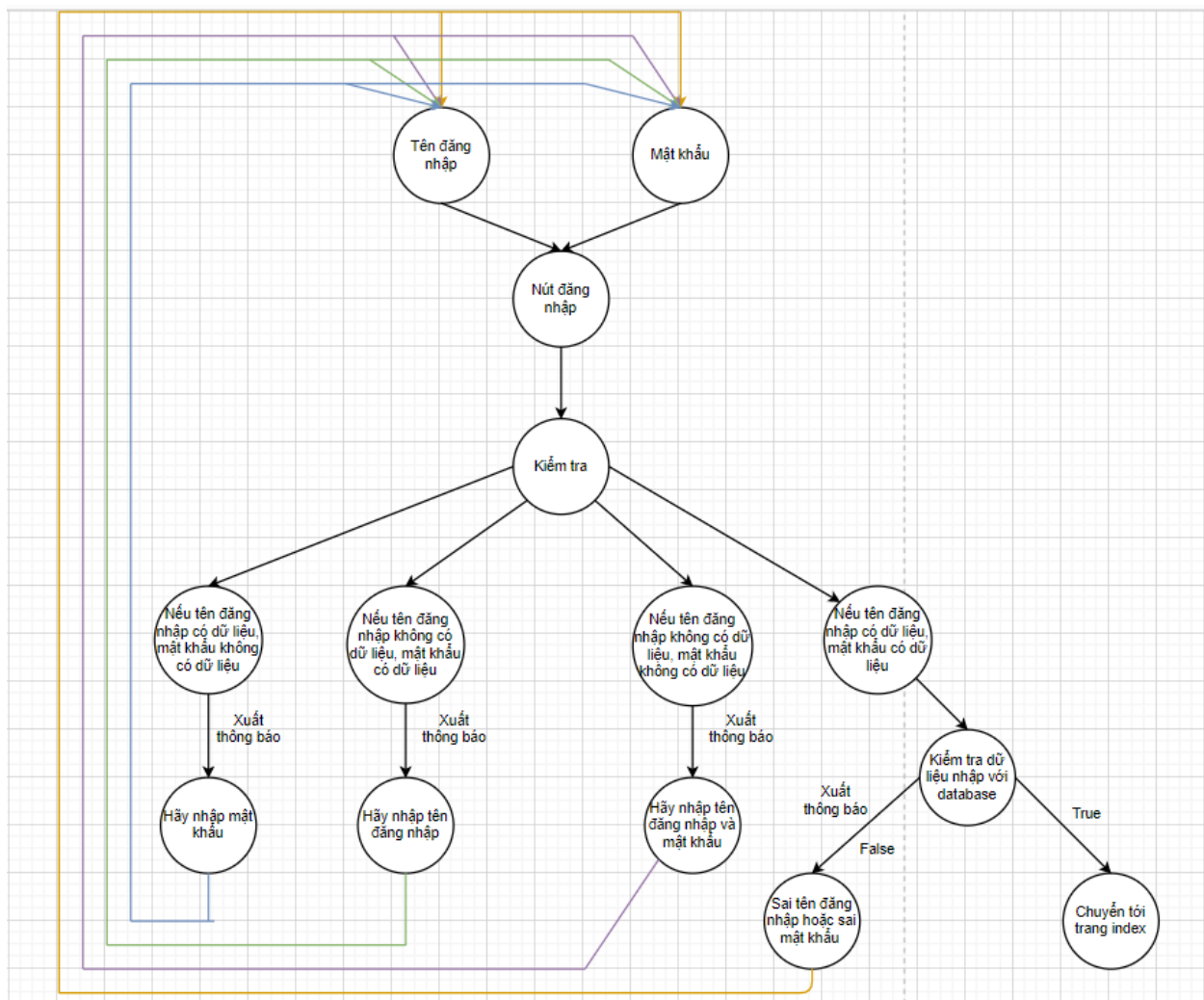
+Đoạn “Ngày 3”: 12/04/2021(0h1p) -> 13/04/2021(23h59p)

+Đoạn “Ngày 4”: 13/04/2021(0h1p) -> 14/04/2021(23h59p)

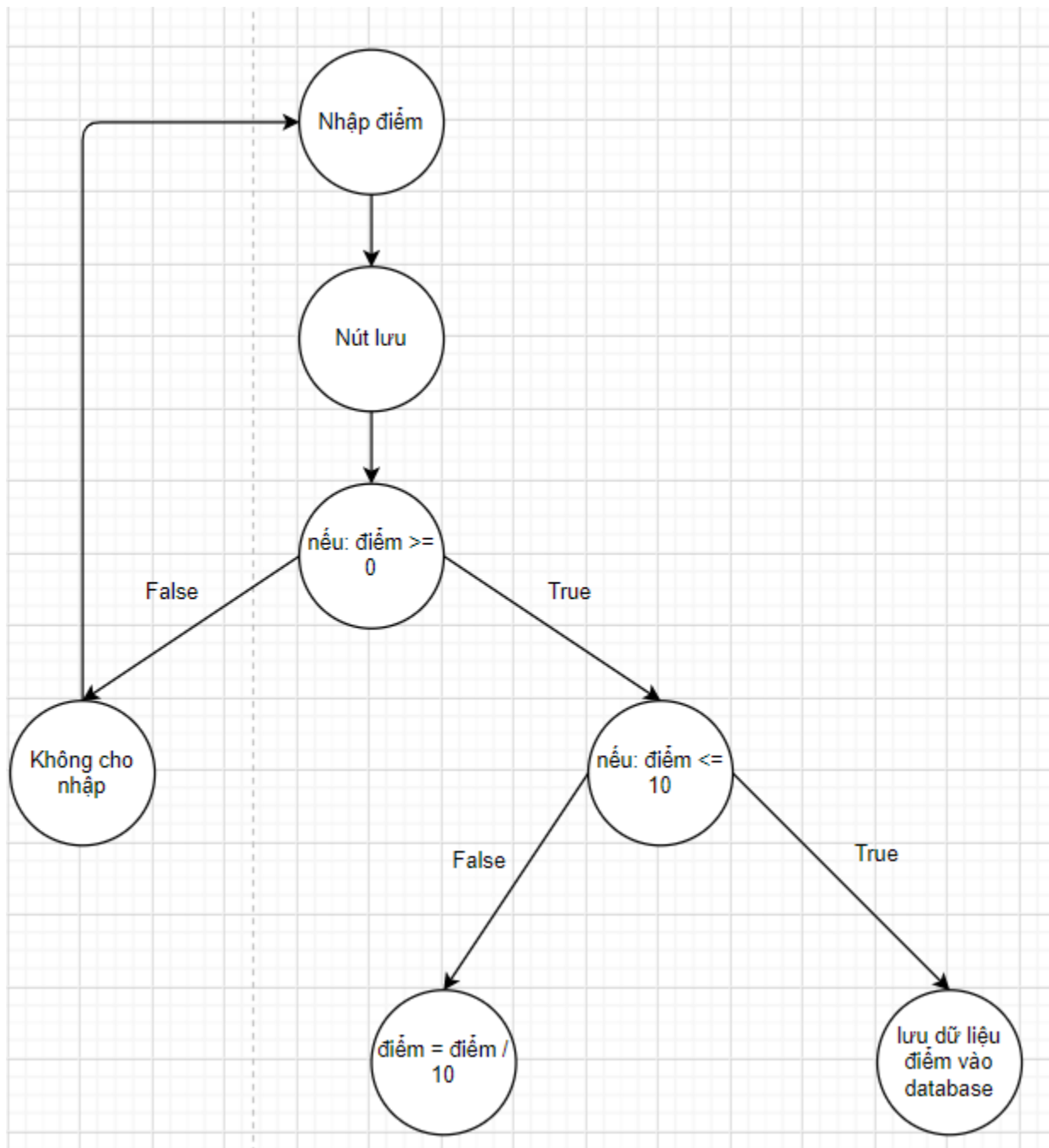
+Đoạn “Ngày 5”: 14/04/2021(0h1p) -> 15/04/2021(23h59p)

+ Đoạn trước ngày 10/04/2021 hoặc sau ngày 15/04/2021 là vùng ngoại biên

-Dịch chuyển trạng thái: chức năng đăng nhập



-Dịch chuyển trạng thái: chức năng nhập điểm giáo viên



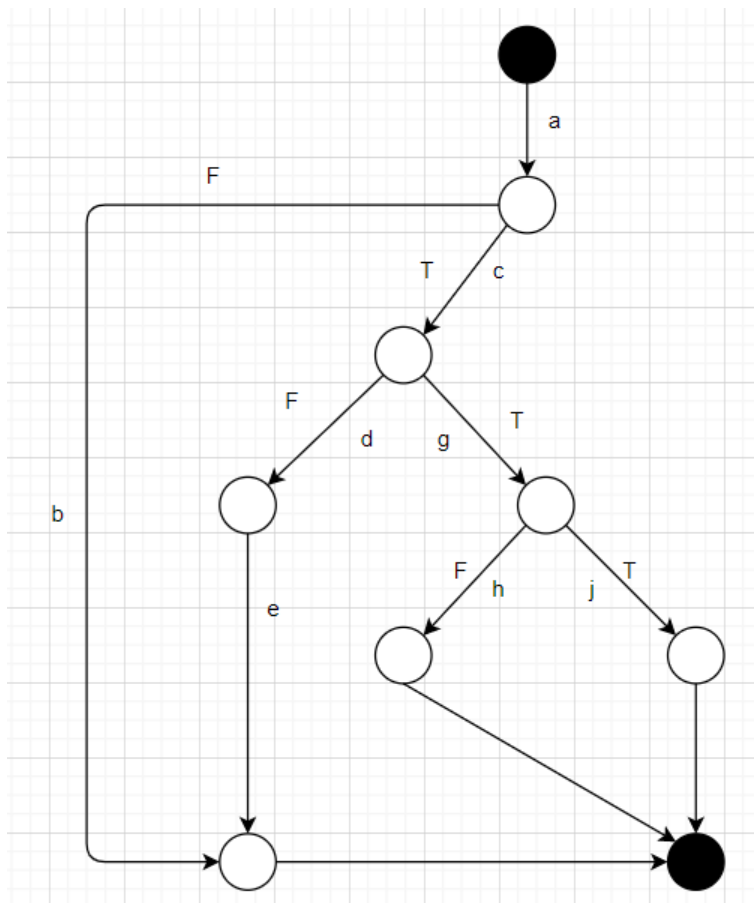
Kiểm thử hộp trắng

-Kiểm thử phủ lệnh

```
[AllowAnonymous]
[HttpPost]
0 references
public async Task<ActionResult> Login(LoginModel lg)
{
    if (ModelState.IsValid)
    {
        DataTable dt = await new TaiKhoanTruongDAL().DangNhap(lg.TaiKhoan, lg.MatKhau);

        if (dt.Rows.Count == 1)
        {
            TK = new TaiKhoanTruong(dt.Rows[0]);
            FormsAuthentication.SetAuthCookie(lg.TaiKhoan, true);
            if (TK.Loai == 1)
            {
                Session["TaiKhoanNhaTruong"] = lg.TaiKhoan.ToString();
                Session["MatKhau"] = lg.MatKhau.ToString();
                mk = lg.MatKhau.ToString();
                return RedirectToAction("Index", "HomeAdmin");
            }
            else
            {
                Session["TaiKhoanGiaoVien"] = lg.TaiKhoan.ToString();
                Session["MatKhau"] = lg.MatKhau.ToString();
                return RedirectToAction("GetTKTruong", "HomeGiaoVien", new { area = "GiaoVien", tenTaiKhoan = lg.TaiKhoan, matKhauTaiKhoan = lg.MatKhau });
            }
        }
        else
        {
            ModelState.AddModelError("", "Tên Đăng Nhập Hoặc Mật Khẩu Không Đúng !");
        }
    }

    return View();
}
```



Đường đi: a, c, g, j

Đầu vào			Đầu ra mong muốn
CheckValue	CheckTK	CheckLoai	
T	admin / admin	1	E1

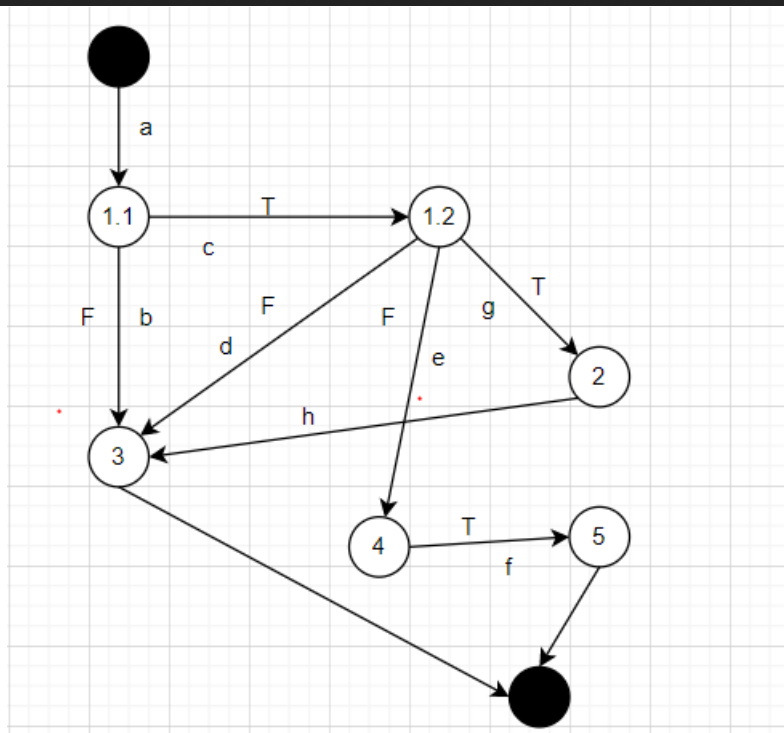
CheckValue: kiểm tra tên đăng nhập và password có dữ liệu không

CheckTK: kiểm tra tên đăng nhập và password có trong database không

CheckLoai: kiểm tra tài khoản thuộc của nhà trường hay của giáo viên

E1: Chuyển tới trang HomeAdmin

```
if (mon != "" && lopDay != "")
{
    if (await tkDal.CheckExist(tenTk) > 0)
    {
        ModelState.AddModelError("", "Tài Khoản Đã Tồn Tại !");
    }
    else
    {
        if (await tkDal.Them(new TaiKhoanTruong(-1, tenTk, tenTk, Loai, tenGV, sdt, int.Parse(mon), int.Parse(lopDay))) != 0)
        {
            return RedirectToAction("Index");
        }
    }
}
else
{
    ModelState.AddModelError("", "Vui Lòng Nhập Đầy Đủ Thông Tin !");
}
```



Đường đi: a, c, e, f

Đầu vào			Đầu ra mong muốn
CheckEmpty	CheckExist	Them	
mon= 'toan' && lopDay= '10C2'	F	T	E1

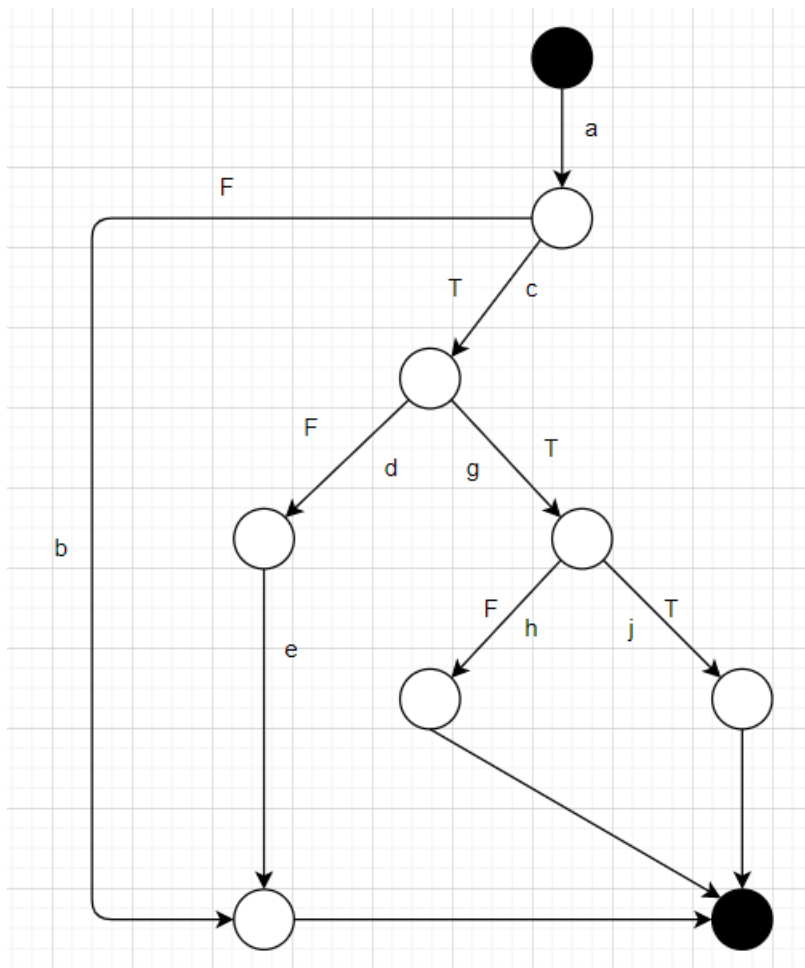
E1: Chuyển tới trang Index

- Kiểm thử nhánh

```
[AllowAnonymous]
[HttpPost]
0 references
public async Task<ActionResult> Login(LoginModel lg)
{
    if (ModelState.IsValid)
    {
        DataTable dt = await new TaiKhoanTruongDAL().DangNhap(lg.TaiKhoan, lg.MatKhau);

        if (dt.Rows.Count == 1)
        {
            TK = new TaiKhoanTruong(dt.Rows[0]);
            FormsAuthentication.SetAuthCookie(lg.TaiKhoan, true);
            if (TK.Loai == 1)
            {
                Session["TaiKhoanNhaTruong"] = lg.TaiKhoan.ToString();
                Session["MatKhau"] = lg.MatKhau.ToString();
                mk = lg.MatKhau.ToString();
                return RedirectToAction("Index", "HomeAdmin");
            }
            else
            {
                Session["TaiKhoanGiaoVien"] = lg.TaiKhoan.ToString();
                Session["MatKhau"] = lg.MatKhau.ToString();
                return RedirectToAction("GetTKTruong", "HomeGiaoVien", new { area = "GiaoVien", tenTaiKhoan = lg.TaiKhoan, matKhauTaiKhoan = lg.MatKhau });
            }
        }
        else
        {
            ModelState.AddModelError("", "Tên Đăng Nhập Hoặc Mật Khẩu Không Đúng !");
        }
    }

    return View();
}
```



Các đường đi phủ nhánh:

+ a, b (F)

+ a, c, g, j (T, T, T)

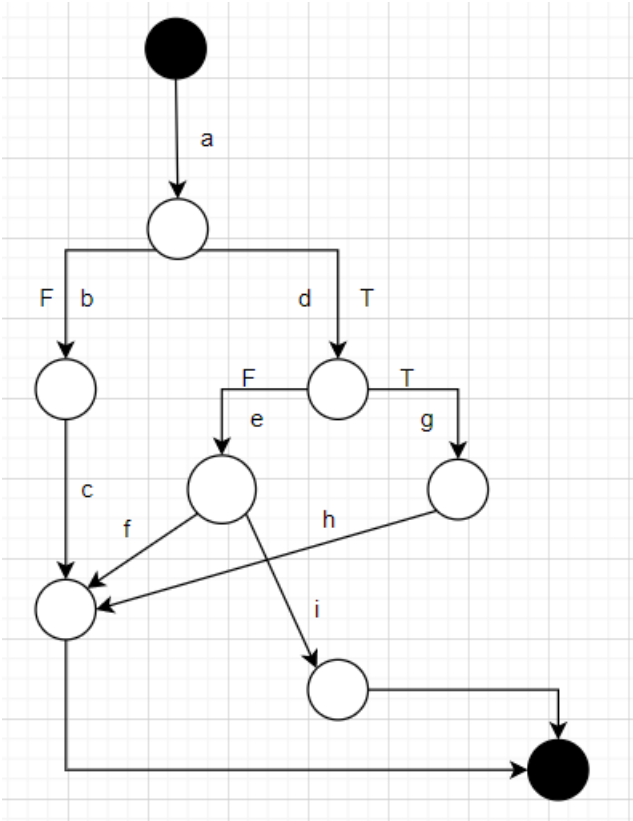
Test case

Đầu vào			Đầu ra mong muốn
CheckValue	CheckTk	CheckLoai == 1	
T	admin/admin	T	E1
F	-	-	E2

E1: Chuyển hướng tới trang HomeAdmin

E2: Vui lòng nhập tên đăng nhập và mật khẩu

```
if (mon != "" && lopDay != "")
{
    if (await tkDal.CheckExist(tenTk) > 0)
    {
        ModelState.AddModelError("", "Tài Khoản Đã Tồn Tại !");
    }
    else
    {
        if (await tkDal.Them(new TaiKhoanTruong(-1, tenTk, tenTk, Loai, tenGV, sdt, int.Parse(mon), int.Parse(lopDay))) != 0)
        {
            return RedirectToAction("Index");
        }
    }
}
else
{
    ModelState.AddModelError("", "Vui Lòng Nhập Đầy Đủ Thông Tin !");
}
```



Các đường đi phủ nhánh:

- + a, b, c
- + a, d, g,h

Test case

Đầu vào			Đầu ra mong muốn
CheckEmpty	CheckTK	Them	
T	admin/admin	-	E1
F	-	-	E3

E1: Tài khoản đã tồn tại

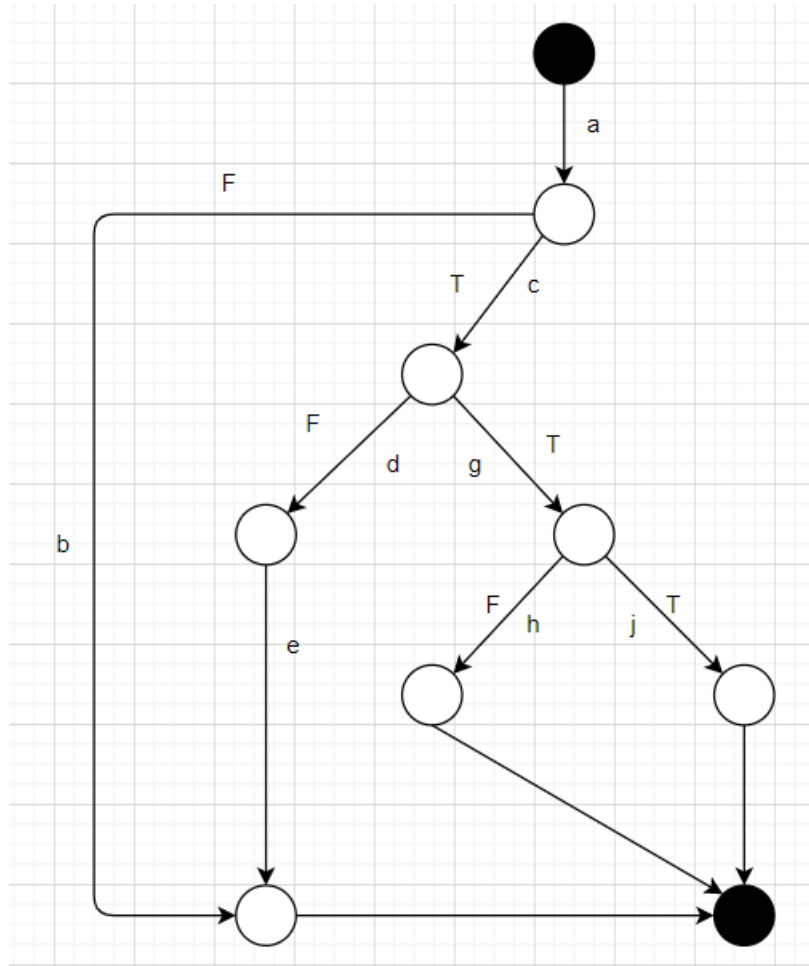
E3: Vui lòng nhập đầy đủ thông tin

-Kiểm thử đường dẫn

```
[AllowAnonymous]
[HttpPost]
0 references
public async Task<ActionResult> Login(LoginModel lg)
{
    if (ModelState.IsValid)
    {
        DataTable dt = await new TaiKhoanTruongDAL().DangNhap(lg.TaiKhoan, lg.MatKhau);

        if (dt.Rows.Count == 1)
        {
            TK = new TaiKhoanTruong(dt.Rows[0]);
            FormsAuthentication.SetAuthCookie(lg.TaiKhoan, true);
            if (TK.Loai == 1)
            {
                Session["TaiKhoanNhaTruong"] = lg.TaiKhoan.ToString();
                Session["MatKhau"] = lg.MatKhau.ToString();
                mk = lg.MatKhau.ToString();
                return RedirectToAction("Index", "HomeAdmin");
            }
            else
            {
                Session["TaiKhoanGiaoVien"] = lg.TaiKhoan.ToString();
                Session["MatKhau"] = lg.MatKhau.ToString();
                return RedirectToAction("GetTKTruong", "HomeGiaoVien", new { area = "GiaoVien", tenTaiKhoan = lg.TaiKhoan, matKhauTaiKhoan = lg.MatKhau });
            }
        }
        else
        {
            ModelState.AddModelError("", "Tên Đăng Nhập Hoặc Mật Khẩu Không Đúng !");
        }
    }

    return View();
}
```



Các đường đi phủ đường dẫn:

+ a, b (F)

+ a, c, d, e (T, F)

+ a, c, g, h (T, T, F)

+ a, c, g, j (T, T, T)

Test case

Đầu vào			Đầu ra mong muốn
CheckValue	CheckTk	CheckLoai == 1	
T	admin/admin	T	E1
T	phong/phong	F	E2
T	dia/dia	-	E3
F	-	-	E4

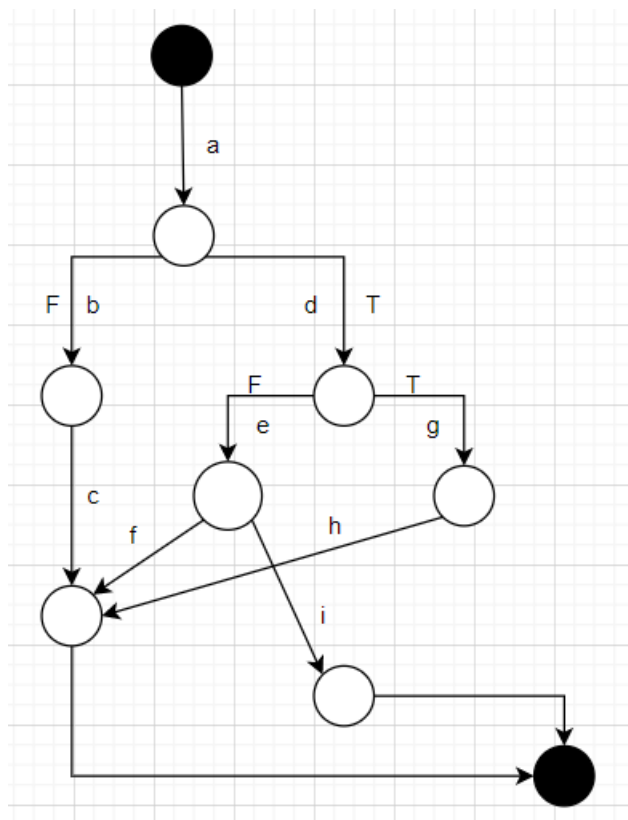
E1: Chuyển hướng tới trang HomeAdmin

E2: Chuyển hướng tới trang HomeGiaoVien

E3: Tên đăng nhập hoặc mật khẩu không đúng

E4: Vui lòng nhập tên đăng nhập và mật khẩu

```
if (mon != "" && lopDay != "")
{
    if (await tkDal.CheckExist(tenTk) > 0)
    {
        ModelState.AddModelError("", "Tài Khoản Đã Tồn Tại !");
    }
    else
    {
        if (await tkDal.Them(new TaiKhoanTruong(-1, tenTk, tenTk, Loai, tenGV, sdt, int.Parse(mon), int.Parse(lopDay))) != 0)
        {
            return RedirectToAction("Index");
        }
    }
}
else
{
    ModelState.AddModelError("", "Vui Lòng Nhập Đầy Đủ Thông Tin !");
}
```



Các đường đi phủ đường dẫn:

+ a, b, c

+ a, d, e, f

+ a, d, e, i

+ a, d, g, h

Test case

Đầu vào			Đầu ra mong muốn
CheckEmpty	CheckTK	Them	
T	admin/admin	-	E1
T	dia/dia	T	E2
F	-	-	E3

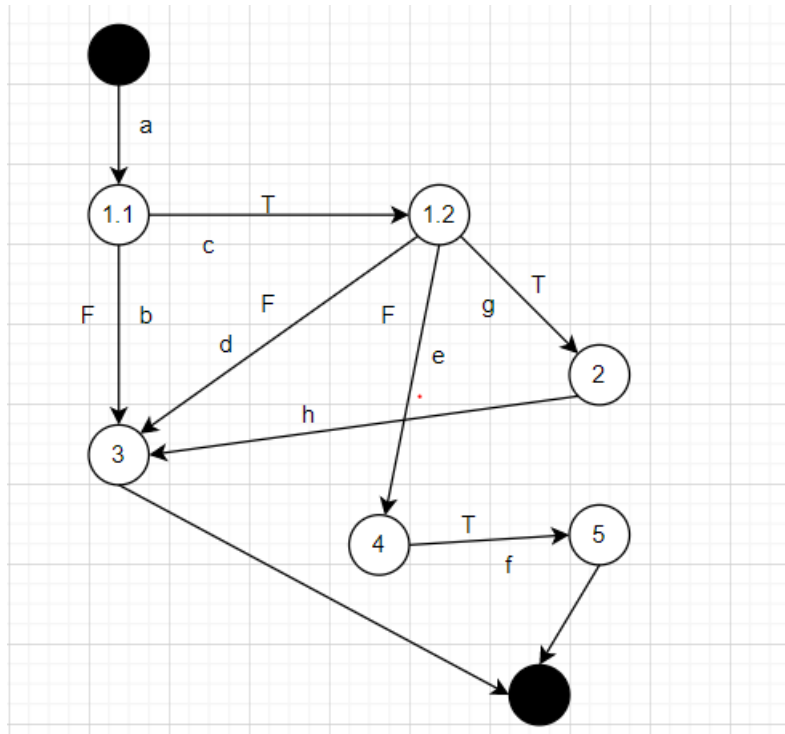
E1: Tài khoản đã tồn tại

E2: Chuyển hướng tới Index

E3: Vui lòng nhập đầy đủ thông tin

-Kiểm thử điều kiện

```
if (mon != "" && lopDay != "")
{
    if (await tkDal.CheckExist(tenTk) > 0)
    {
        ModelState.AddModelError("", "Tài Khoản Đã Tồn Tại !");
    }
    else
    {
        if (await tkDal.Them(new TaiKhoanTruong(-1, tenTk, tenTk, Loai, tenGV, sdt, int.Parse(mon), int.Parse(lopDay))) != 0)
        {
            return RedirectToAction("Index");
        }
    }
}
else
{
    ModelState.AddModelError("", "Vui Lòng Nhập Đầy Đủ Thông Tin !");
}
```



Các đường đi:

+ a, b

+ a, c, d

+ a, c, e, f

+ a, c, g, h

Test case

Đường dẫn	Đầu vào			Đầu ra mong muốn
	mon	lopDay	tenTK	
a, b	-	-	-	E1
a, c, d	hoa	-	-	E2
a, c, e, f	hoa	10A2	Thanh	E3
a, c, g, h	hoa	10A2	Phong	E4

-E1: vui lòng nhập đầy đủ thông tin

-E2: lớp dạy không bỏ trống

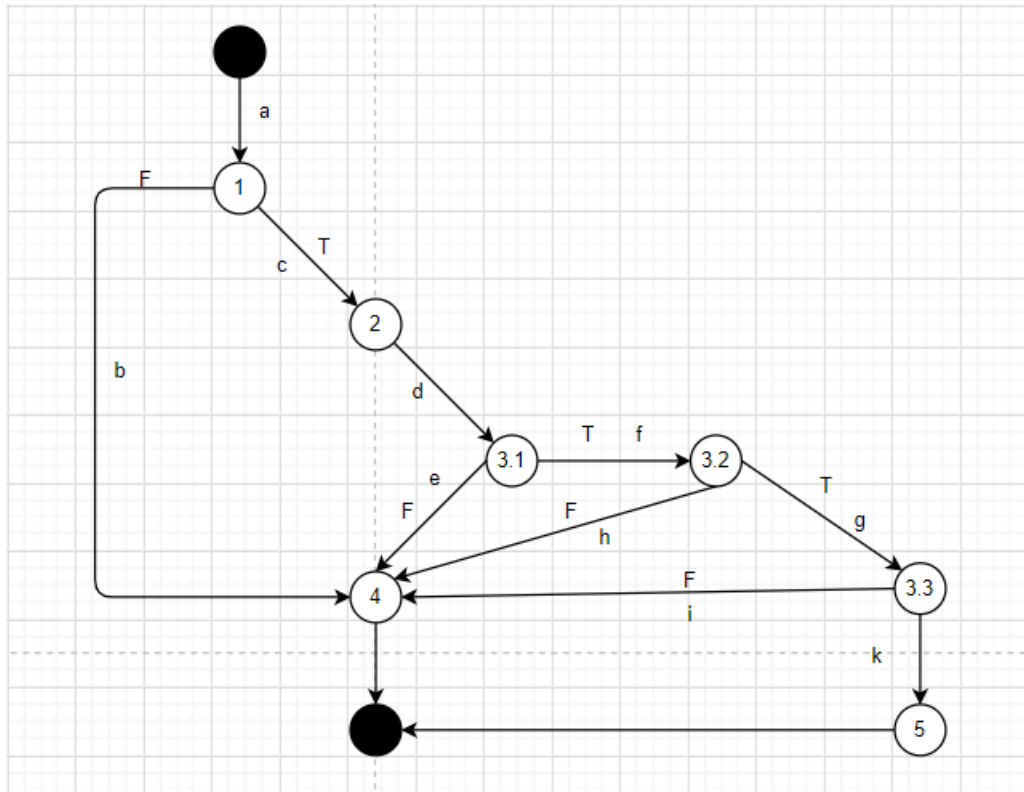
-E3: Chuyển trang tới Index

-E4: Tài Khoản đã tồn tại

```

public async Task<ActionResult> ThemHS_ByExcel(ImportExcel importExcel)
{
    try
    {
        if (ModelState.IsValid)
        {
            if (Request != null)
            {
                DataTable dt = new DataTable();
                HttpPostedFileBase file = importExcel.file;
                if ((file != null) && (file.ContentLength > 0) && !string.IsNullOrEmpty(file.FileName))
                {
                    string fileName = file.FileName;
                    string path = Server.MapPath("~/Content/Upload/" + fileName);
                    file.SaveAs(path);
                    var excelData = new ExcelData(path);
                    var sData = excelData.getData("Sheet1");
                    List<ThongTinHS> list = new List<ThongTinHS>();
                    dt = sData.CopyToDataTable();
                    foreach (DataRow item in dt.Rows)
                    {
                        ThongTinHS hs = new ThongTinHS();
                        await new ThongTinHSDAL().ThemTTHS(
                            hs.Ten = item["Tên"].ToString(),
                            hs.NgaySinh = Convert.ToDateTime(item["Ngày Sinh"]),
                            hs.GioiTinh = Convert.ToByte(item["Giới Tính"]),
                            hs.NoiSinh = item["Nơi Sinh"].ToString(),
                            hs.DanToc = item["Dân Tộc"].ToString(),
                            hs.TonGiao = item["Tôn Giáo"].ToString(),
                            hs.IDlop = Convert.ToInt32(item["Mã Lớp"]),
                            hs.IDloaiHocSinh = Convert.ToInt32(item["Mã Chương Trình Học"])
                        );
                    }
                    return RedirectToAction("LoadTable", "QuanLyThongTinHS", new { id = lop.ID });
                }
            }
        }
    }
    catch (Exception e)
    {
        Console.WriteLine(e);
        ViewBag.Loai = "Nhập dữ liệu thất bại ,Kiểm Tra Lại Định Dạng File Excel !";
    }
    return View();
}

```



Các đường đi:

+ a, b

+ a, c, d, e

+ a, c, d, f, h

+ a, c, d, f, g, k

+ a, c, d, f, g, i

Test case

Đường dẫn	Đầu vào					Đầu ra mong muốn
	CheckValue	CheckRes	CheckFile	CheckF.Length	CheckEmpty	
a, b	F	-	-	-	-	E1
a, c, d, e	T	T	F	-	-	E1
a, c, d, f, h	T	T	T	F	-	E1
a, c, d, f, g, i	T	T	T	T	F	E1
a, c, d, f, g, k	T	T	T	T	T	E2

CheckValue: ModelState.Isvalue

CheckRes : Resquest != null

CheckFile: File != null

CheckF.length: file.ContentLength > 0

CheckEmpty: !string.IsNullOrEmpty(file.FileName)

E1: Nhập dữ liệu thất bại

E2: Chuyển hướng tới trang quản lý thông tin sinh viên

Kiểm Thử Lướt Tải Với Công Cụ LoadStorm

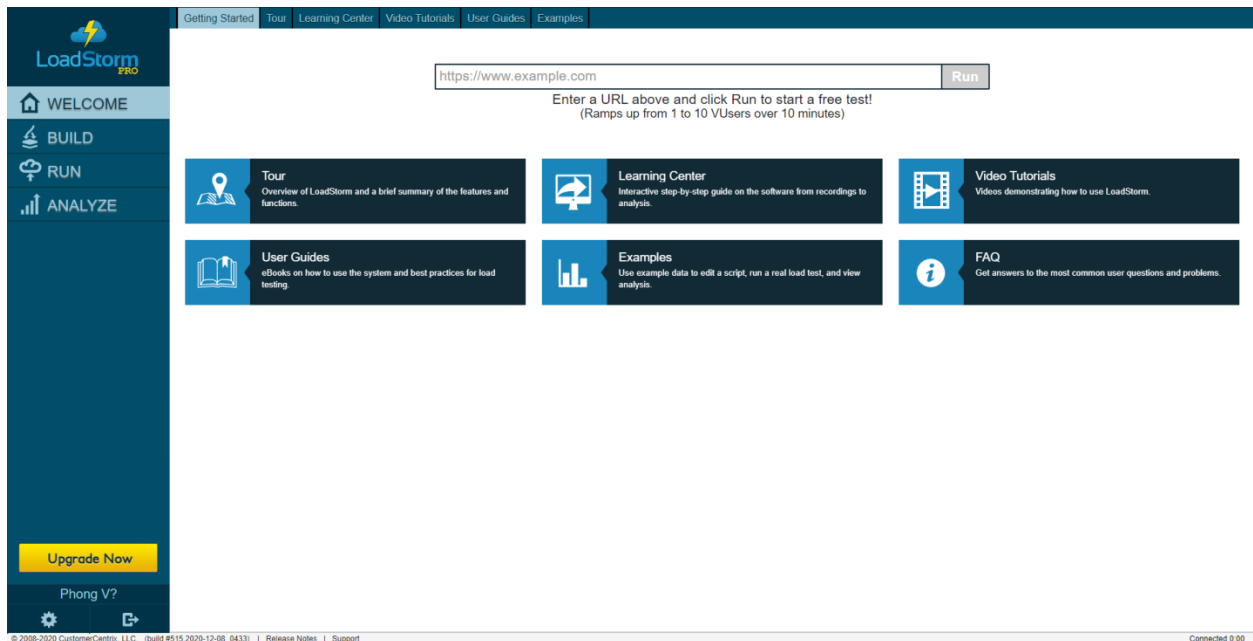
Giới thiệu

LoadStorm là một công cụ kiểm thử cloudtest cho các ứng dụng web và mobile. Điểm mạnh là nó có thể kiểm tra hiệu năng của ứng dụng dựa trên lượng user và traffic. LoadStorm cũng có khả năng chịu tải rất tốt khi mà nó có thể giả lập hàng trăm nghìn đến hàng triệu user để tìm kiếm các breaking point trong ứng dụng. Các kịch bản test(scenario) của LoadStorm có thể được chỉnh sửa bởi tester.

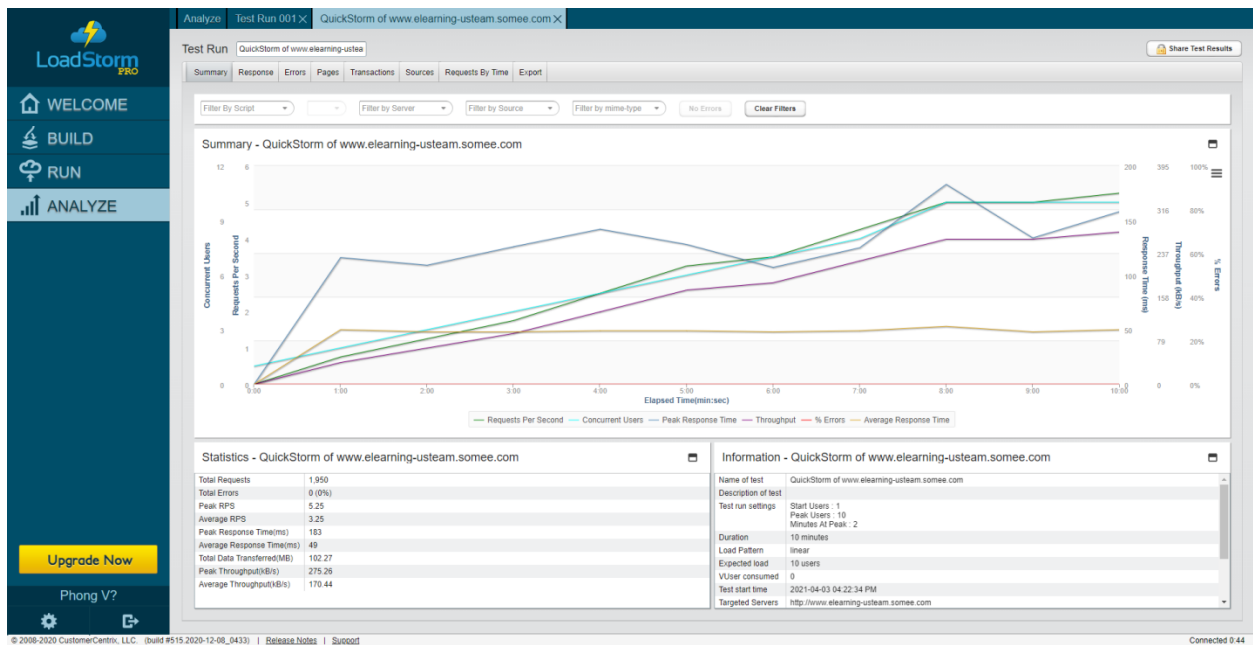
Testing với Loadstorm

Để test với Loadstorm thì người dùng cần đăng ký tài khoản, Loadstorm hiện có phiên bản free và pro tại địa chỉ <https://loadstorm.com/>

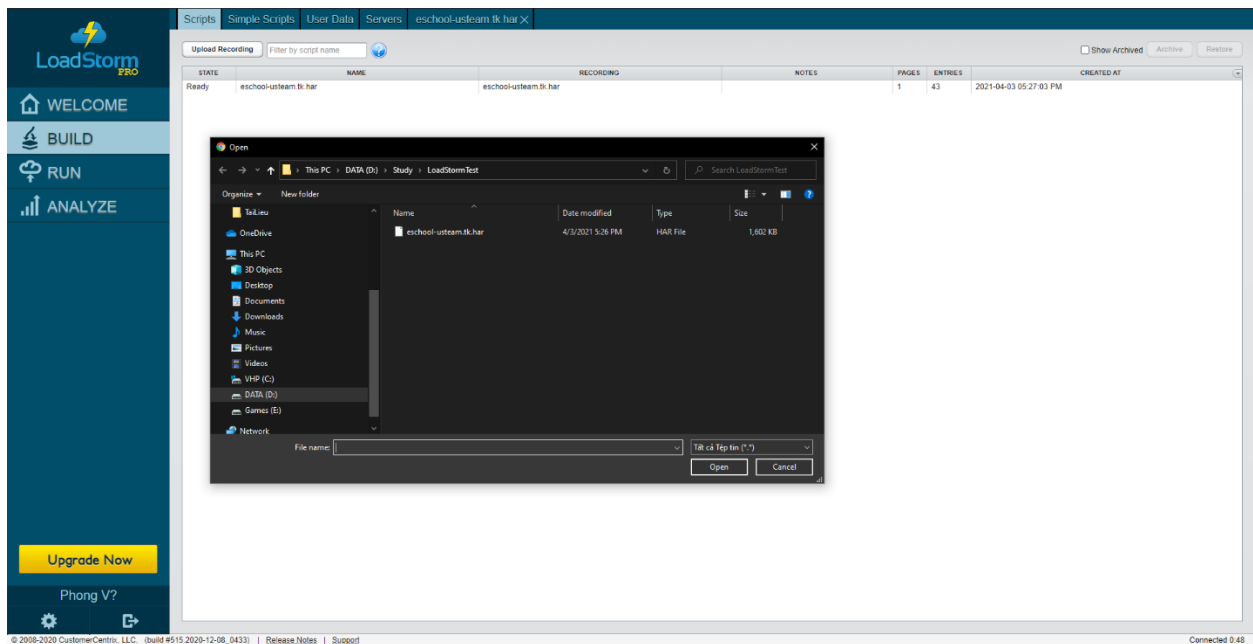
Sau khi đã có tài khoản đăng nhập vào trang chủ sẽ có các chức năng



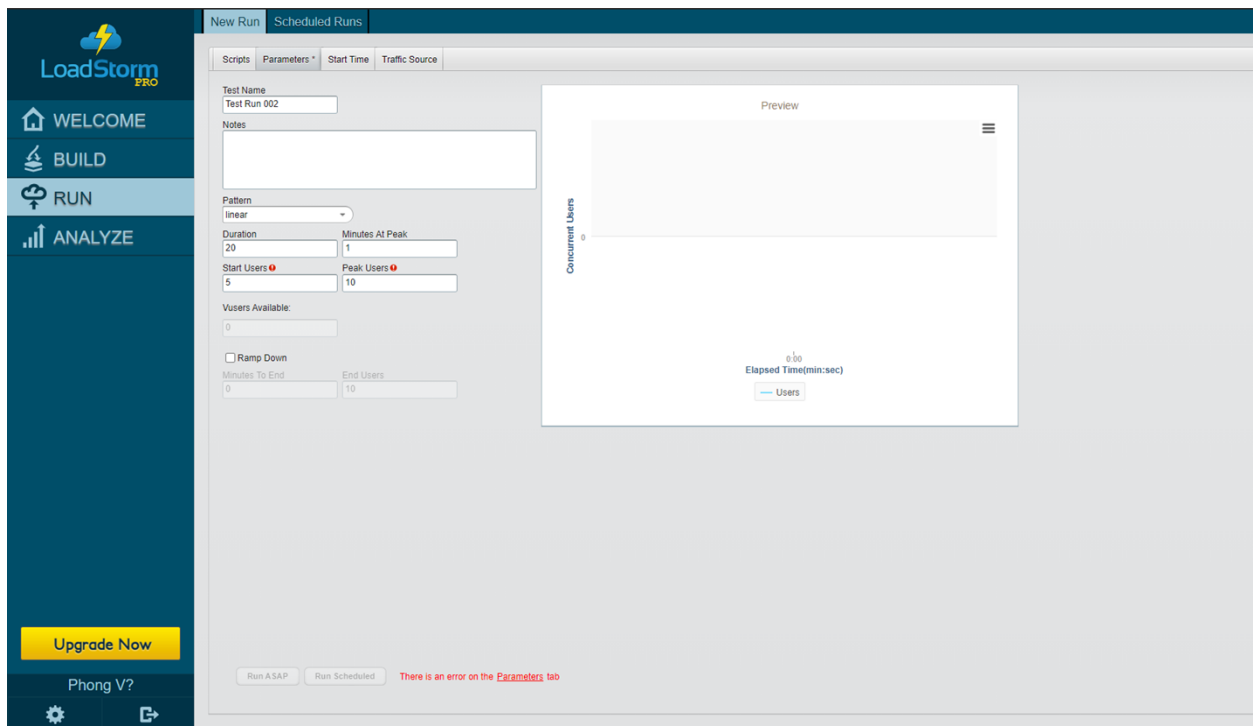
Tại màn hình welcome Loadstorm sẽ cho phép làm quick test chỉ cần nhập url trang web cần test Loadstorm sẽ thực hiện loadtest trang web đó khoảng 10 phút. Sau thời gian đó hệ thống Loadstorm sẽ hiển thị các thông số.



Loadstorm cũng sẽ cho phép tester nhập các kịch bản để hệ thống có thể test ở tab build. Bằng cách thêm file har vào phần upload. File har là file hành động của user sau 1 kịch bản mà được soạn trước.

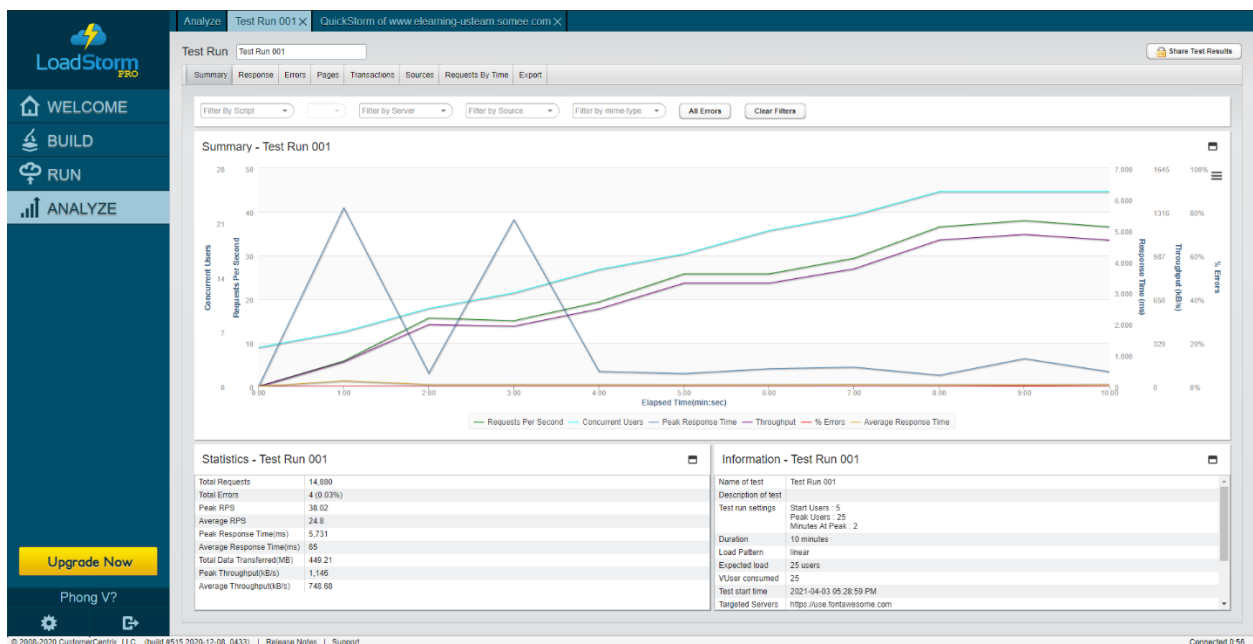


LoadStorm sẽ cho người dùng điều chỉnh các thông số và sẽ bắt đầu test ở tab run.



Ở tab này Loadstorm sẽ cho người dùng cài đặt các thông số Pattern, Duration, Minute At Peak, Start Users, Peak Users.

Sau khi cài đặt các thông số và cho chạy hệ thống sẽ có bảng phân tích đánh giá trực quan, ở đây tôi sẽ test hoạt cảnh chức năng đăng nhập của 1 trang web. Sau khi upload file har và cài đặt sẽ thu được kết quả.



Các thông số trên bảng kết quả cho thấy:

Tổng lượng request trong 10p: 14,880

Tổng lỗi: 4

Đỉnh điểm request: 38.02

Trung bình lượng request: 24.8

Thời gian đợi request/s: 5.731ms

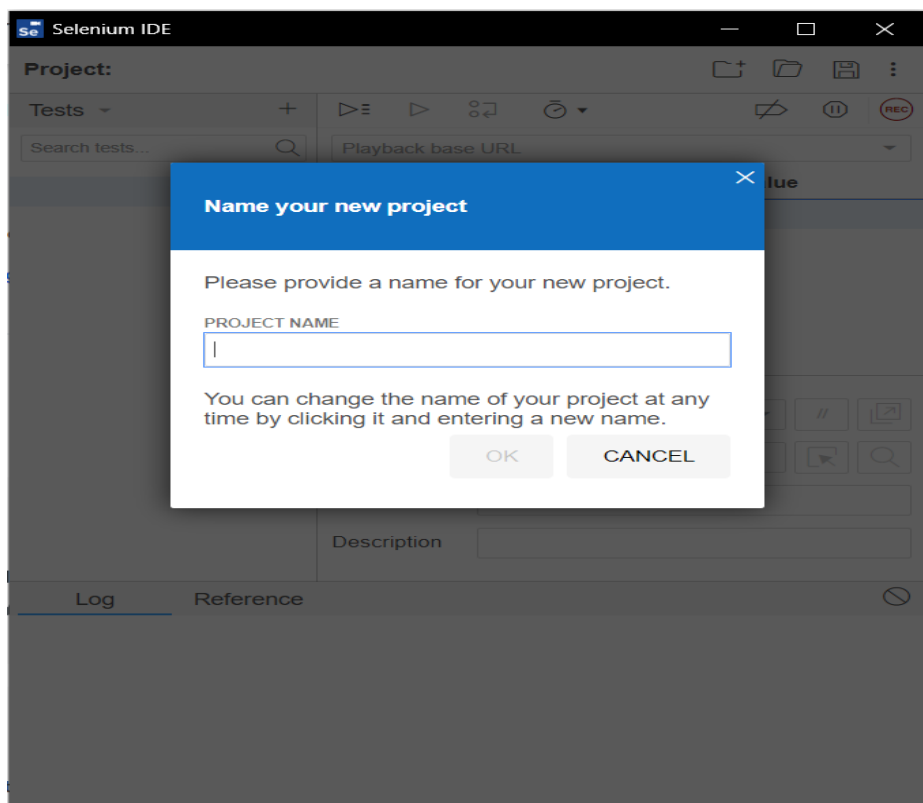
Thời gian phản hồi trung bình: 65ms

Tổng cộng lượng data truyền: 449.21 mb

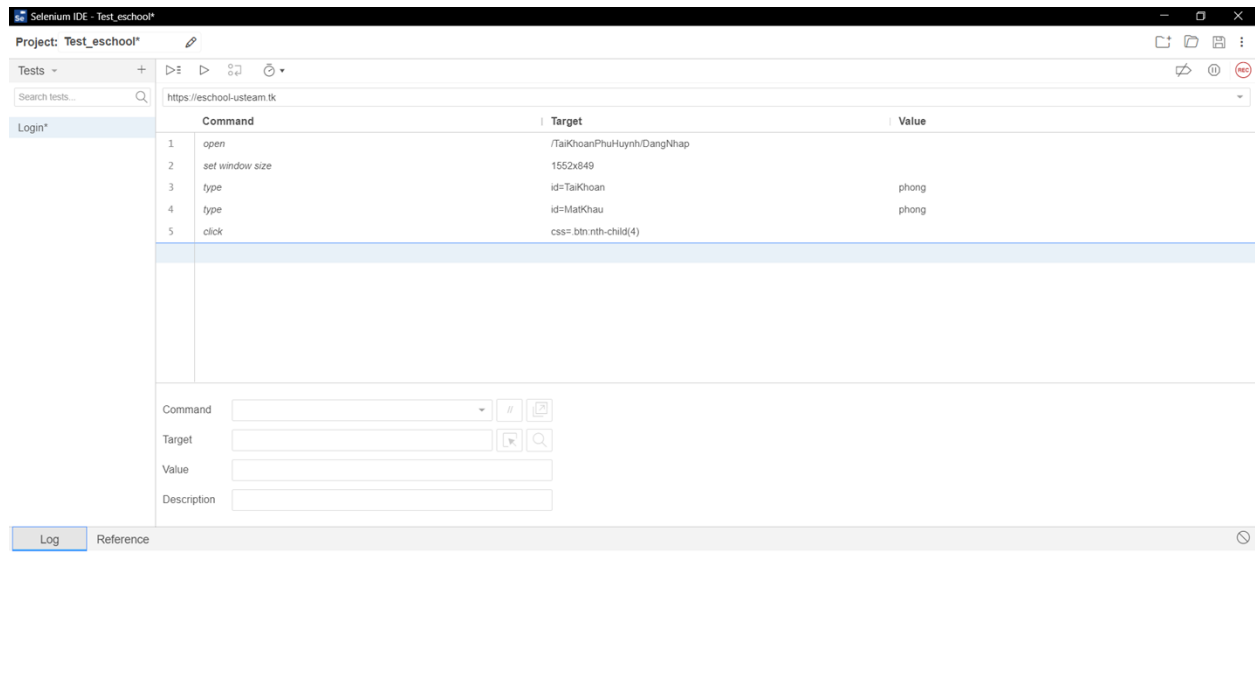
Kiểm Thử Website với công cụ Selenium

Selenium IDE (Integrated Development Environment) là công cụ mẫu cho việc xây dựng kịch bản kiểm thử. Nó là một plugin của Firefox và cung cấp một giao diện để sử dụng cho sự phát triển kiểm thử tự động. Selenium IDE có tính năng ghi âm, ghi lại hoạt động người dùng, những gì họ thực hiện, sau đó xuất chúng dưới dạng dưới dạng một tập lên trong một hoặc nhiều ngôn ngữ chương trình sẽ thực thi sau. Note: Mặc dù Selenium IDE có tính năng “Save” cho phép người dùng lưu giữ lần kiểm tra trong một định dạng bảng để import vào sau và thực thi. Nó không phải được thiết kế để chạy cho lần kiểm tra của bạn đạt yêu cầu mà nó được xây dựng tất cả lần kiểm tra tự động hóa mà bạn sẽ phải làm. Đặc biệt, Selenium IDE không cung cấp sự lặp lại hoặc câu lệnh có điều kiện cho tập lệnh kiểm tra. Vào thời điểm viết chúng thì không có kế hoạch thêm điều đó. Lý do là một phần kỹ thuật hoặc một phần cơ bản dựa trên các nhà phát triển Selenium khuyến khích thực tiễn tốt nhất trong tự động hóa thử nghiệm, điều mà luôn luôn đòi hỏi một số lượng chương trình. Selenium IDE chỉ đơn giản là một công cụ tạo mẫu nhanh chóng. Các nhà phát triển Selenium đề nghị cho nghiêm trọng, tự động kiểm tra mạnh mẽ Selenium 2 hoặc Selenium 1 trong nhiều ngôn ngữ lập trình hỗ trợ nhiều.

Bắt đầu vào giao diện chính của selenium và chúng ta sẽ tạo 1 project mới



Sau khi tạo project và start 1 record chúng ta sẽ vào được màn hình chính của công cụ selenium



Test dữ liệu kiểm thử đăng nhập trang eschool-us.tk

Bảng dữ liệu kiểm thử

Stt	Tên đăng nhập	Mật khẩu	Kết quả dự kiến	Kết quả thực tế	Trạng thái
1	phong	phong	Đăng nhập thành công	Đăng nhập thành công	đúng
2	phong	Trống	Nhập mật khẩu	Thông báo nhập mật khẩu	đúng
3	Trống	Trống	Thông báo nhập tên và mật khẩu	Thông báo nhập tên và mật khẩu	đúng
4	phong	Mật khẩu sai	Thông báo sai tài khoản	Thông báo sai tài khoản	Đúng

Stt: 1

Project: Test_eschool*

Tests

Search tests...

Run current test Ctrl+R k

	Command	Target	Value
DangNhapPH*	1 open	/TaiKhoanPhuHuynh/DangNhap	
	2 set window size	1552x849	
	3 type	id=TaiKhoan	phong
	4 type	id=MatKhau	phong
	5 click	css= fa-sign-in-alt	

Command

Target

Value

Description

Log

Reference

Running 'DangNhapPH'

1. open on /TaiKhoanPhuHuynh/DangNhap OK

2. setWindowSize on 1552x849 OK

3. type on id=TaiKhoan with value phong OK

4. type on id=MatKhau with value phong OK

5. click on css= fa-sign-in-alt OK

'DangNhapPH' completed successfully

09:15:07

09:15:09

09:15:09

09:15:09

09:15:11

09:15:12

09:15:13

Stt: 2

US TEAM

phong

Mật Khẩu

Hãy Nhập Mật Khẩu !

Đăng Nhập

Hệ Thống Tài Liệu

Quên Mật Khẩu ?

Project: Test_eschool*

Tests

Search tests...

Run current test Ctrl+R k

	Command	Target	Value
DangNhapPH*	1 open	/TaiKhoanPhuHuynh/DangNhap	
	2 set window size	1552x849	
	3 type	id=TaiKhoan	phong
	4 type	id=MatKhau	
	5 click	css= fa-sign-in-alt	

Command

Target

Value

Description

Log

Reference

Running 'DangNhapPH'

1. open on /TaiKhoanPhuHuynh/DangNhap OK

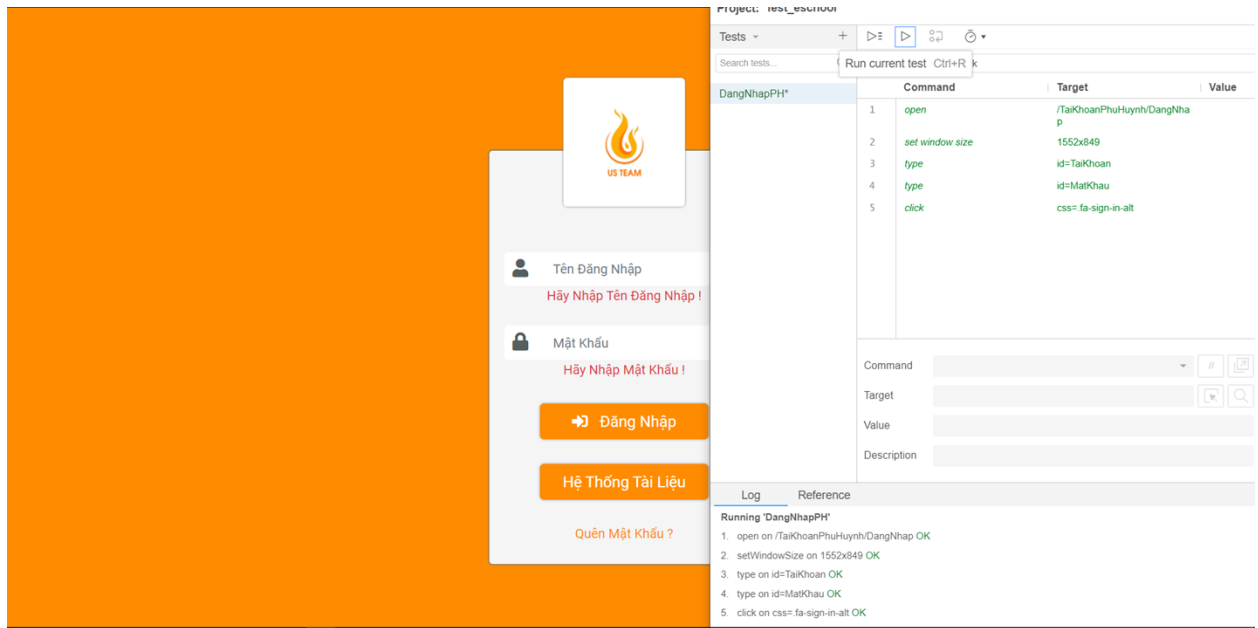
2. setWindowSize on 1552x849 OK

3. type on id=TaiKhoan with value phong OK

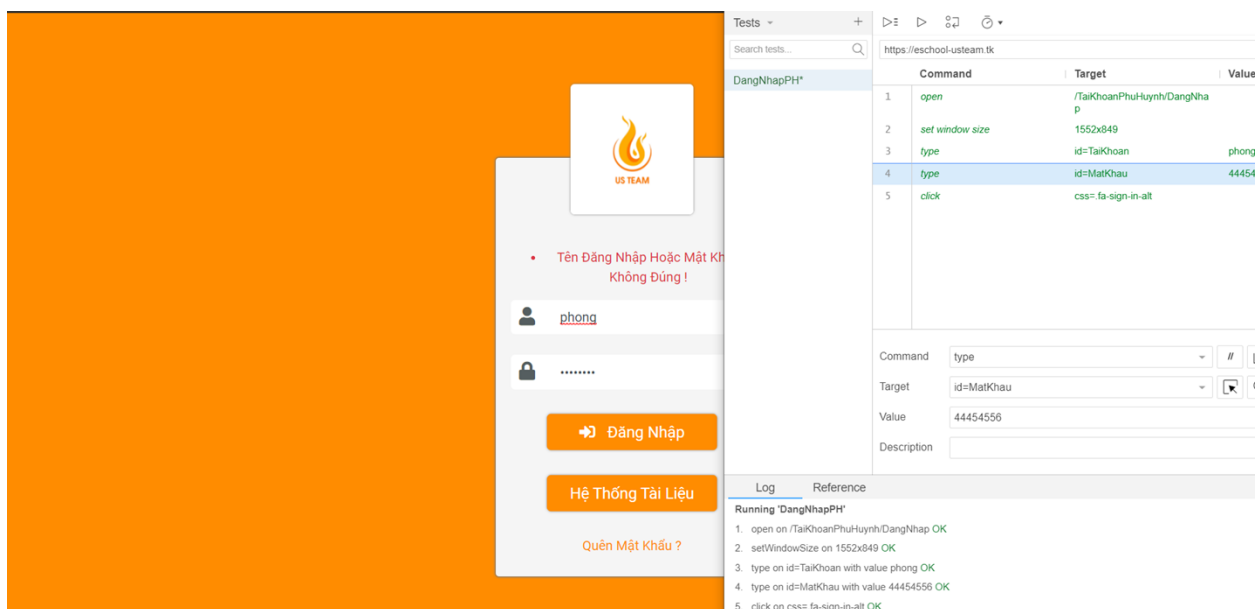
4. type on id=MatKhau OK

5. click on css= fa-sign-in-alt OK

Stt: 3



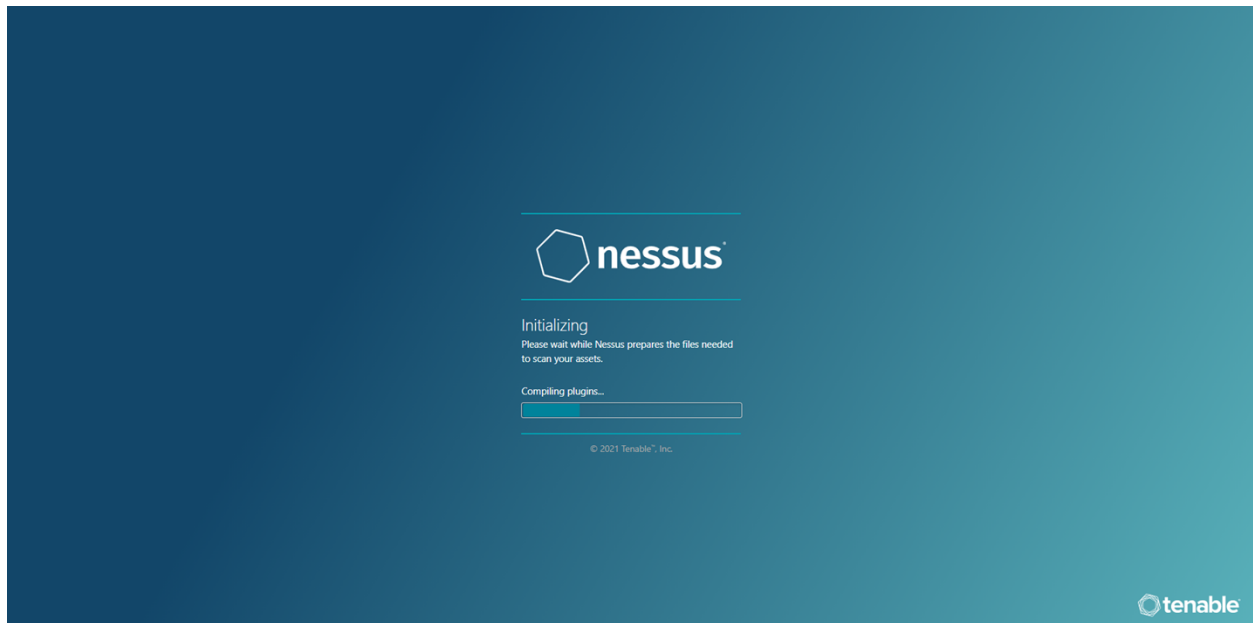
Stt: 4



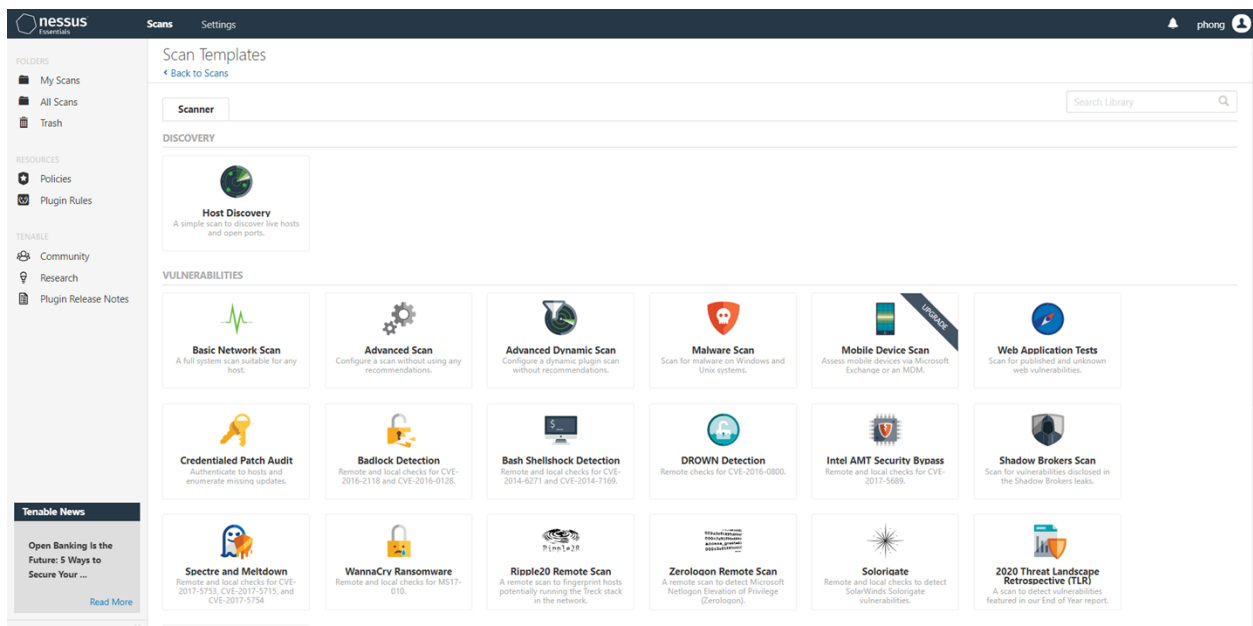
Công Cụ Nessus

Nessus là công cụ chuyên dùng để Pentest hệ thống, rà quét lỗ hổng bảo mật, và mã độc. Công cụ này cho phép bạn dùng thử miễn phí. Điểm khác biệt ở công cụ này so với các công cụ Pentest khác là Nessus chứa các plugin về bảo mật hàng đầu thế giới, có thể rà quét lỗ hổng trong windows, Linux một cách toàn diện.

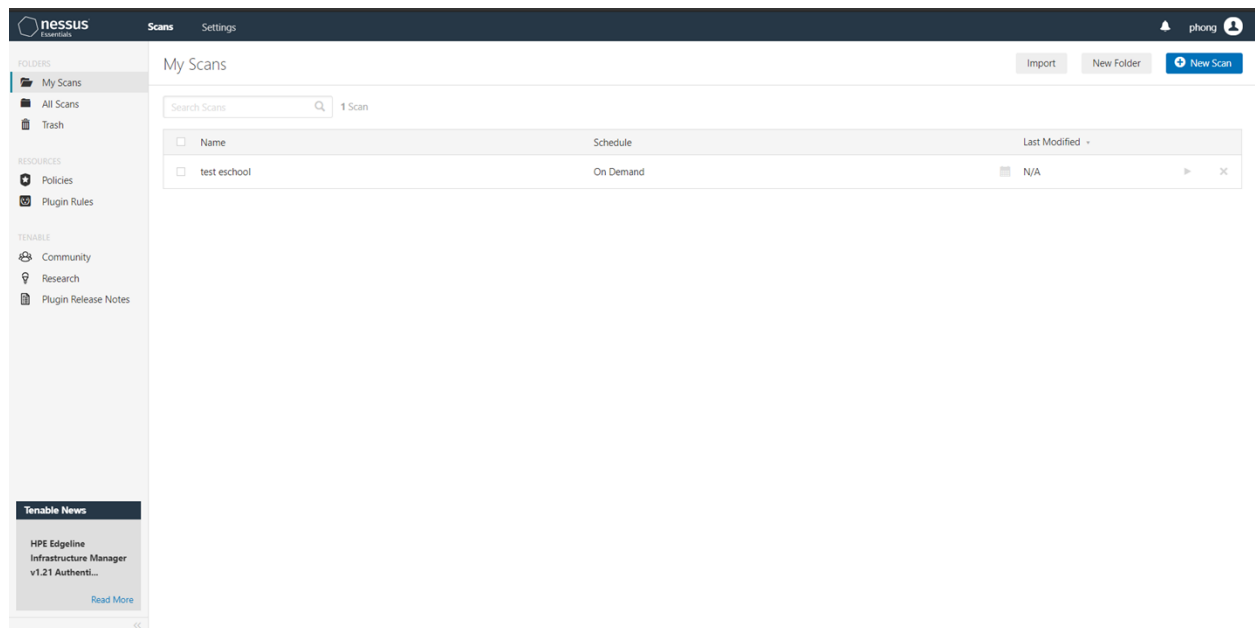
Đăng ký tài khoản hệ thống Nessus và bắt đầu cài đặt



Giao diện menu công cụ



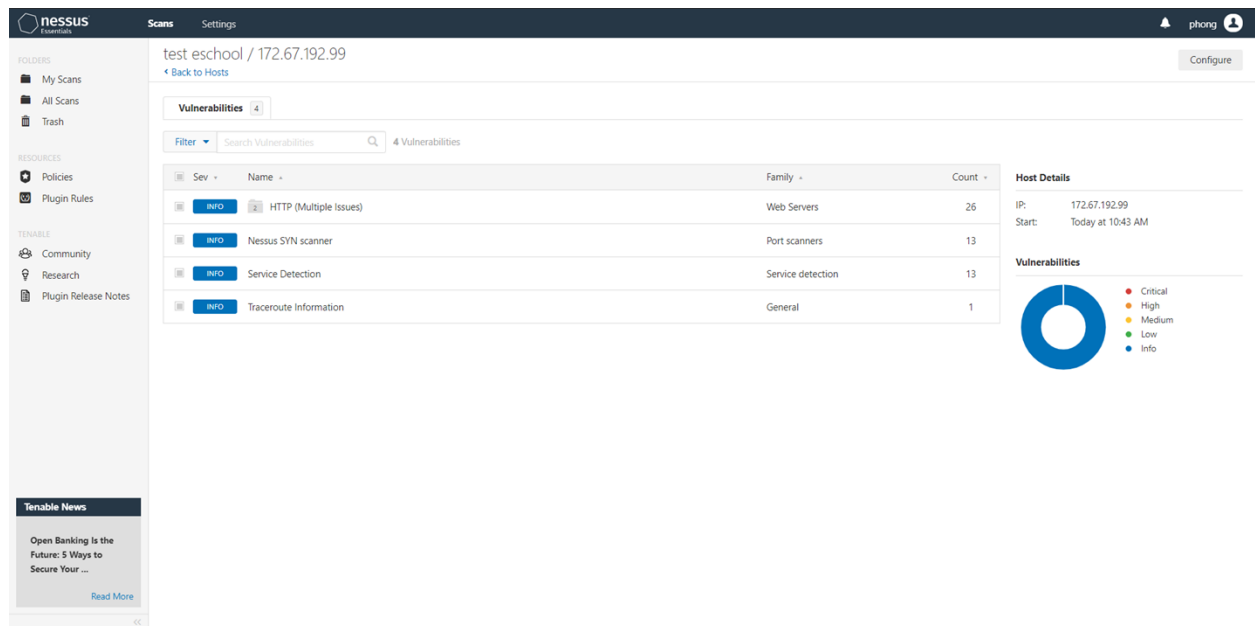
Add địa chỉ ip trang web vào và bắt đầu scan



The screenshot shows the Nessus Scans page. The left sidebar contains navigation links for FOLDERS (My Scans, All Scans, Trash), RESOURCES (Policies, Plugin Rules), and TENABLE (Community, Research, Plugin Release Notes). The main content area is titled 'My Scans' and includes a search bar, a table with one scan entry, and buttons for Import, New Folder, and New Scan. The scan entry is 'test eschool' with a schedule of 'On Demand' and a last modified date of 'N/A'.

Name	Schedule	Last Modified
test eschool	On Demand	N/A

Công cụ sẽ scan các lỗ hổng bảo mật và hiển thị bên tab Vulnerabilities



The screenshot shows the Nessus Vulnerabilities page for the scan 'test eschool'. The left sidebar is the same as the previous screenshot. The main content area is titled 'test eschool / 172.67.192.99' and includes a 'Back to Hosts' link, a 'Configure' button, and a table of vulnerabilities. The table has columns for Severity, Name, Family, and Count. There are 4 vulnerabilities listed. To the right of the table is a 'Host Details' section showing IP, Start time, and a 'Vulnerabilities' donut chart.

Sev	Name	Family	Count
INFO	HTTP (Multiple Issues)	Web Servers	26
INFO	Nessus SYN scanner	Port scanners	13
INFO	Service Detection	Service detection	13
INFO	Traceroute Information	General	1

Host Details

IP: 172.67.192.99
Start: Today at 10:43 AM

Vulnerabilities

Donut chart showing the distribution of vulnerabilities by severity: Critical (red), High (orange), Medium (yellow), Low (green), Info (blue).