

# LTHD - Final Project - Internet Banking

tags: **WNC-LT** **project**

**Yêu cầu:** xây dựng ứng dụng web **Internet Banking**

## 1. Phân hệ khách hàng **customer**

### 1.1 Đăng nhập

- Có sử dụng Google Recaptcha

### 1.2 Liệt kê danh sách tài khoản của người dùng

- Mỗi tài khoản chỉ thể hiện số-tài-khoản & số-dư-hiện-tại
- Mỗi tài khoản chỉ có duy nhất 01 **tài-khoản-thanh-toán**

### 1.3 Thiết lập danh sách người nhận

- Thông tin gồm { số-tài-khoản, tên-gợi-nhớ }.
- Trong trường hợp người dùng không nhập tên-gợi-nhớ, hệ thống sử dụng tên-đăng-ký của tài khoản người nhận làm tên-gợi-nhớ
- Hệ thống hỗ trợ khách hàng quản lý danh sách người nhận này (loại bỏ, điều chỉnh thông tin, ...)

### 1.4 Chuyển khoản

#### 1.4.1 Chuyển khoản nội bộ (cùng ngân hàng)

- Chọn tài-khoản-thanh-toán nguồn
- Điền thông tin người nhận
  - TH người nhận nằm ngoài danh sách: điền số tài khoản, hệ thống tự động truy vấn các thông tin còn lại, KHÔNG truy vấn số dư, của người nhận
  - TH người nhận đã có trong danh sách: người gửi có thể chọn người nhận từ danh sách đã lưu
- Nhập số tiền chuyển & nội dung chuyển

- Chọn hình thức thanh toán phí (người nhận trả phí / người gửi trả phí)
- Ra lệnh CHUYỂN TIỀN
- Yêu cầu người gửi nhập OTP xác nhận giao dịch (người gửi nhận OTP qua email sau khi ra lệnh CHUYỂN TIỀN)
- Hệ thống đề nghị người gửi lưu lại thông tin người nhận cho lần giao dịch kế.

#### 1.4.2 Chuyển khoản liên ngân hàng

- Chọn tài-khoản-thanh-toán nguồn
- Điền thông tin người nhận
  - TH người nhận nằm ngoài danh sách: điền số tài khoản & **ngân hàng**, hệ thống tự động truy vấn các thông tin còn lại, KHÔNG truy vấn số dư, của người nhận
  - TH người nhận đã có trong danh sách: người gửi có thể chọn người nhận từ danh sách đã lưu
- Nhập số tiền chuyển & nội dung chuyển
- Chọn hình thức thanh toán phí (người nhận trả phí / người gửi trả phí)
- Ra lệnh CHUYỂN TIỀN
- Yêu cầu người gửi nhập OTP xác nhận giao dịch (người gửi nhận OTP qua email sau khi ra lệnh CHUYỂN TIỀN)
- Hệ thống đề nghị người gửi lưu lại thông tin người nhận cho lần giao dịch kế.

Chỉ có thể chuyển khoản liên ngân hàng được với ngân hàng **đã liên kết**.

**Email OTP** phải có định dạng nghiêm túc (xem ví dụ bên dưới)

Dear Leomar Umpad,

You have selected [\[redacted\]@gmail.com](#) as your new verification page:

**322144**

This code will expire three hours after this email was sent.

**Why you received this email.**

Apple requires verification whenever an email address is added to a new device.

If you did not make this request, you can ignore this email.

## 1.5 Quản lý nhắc nợ

### 1.5.1 Tạo nhắc nợ

- Điền thông tin người nợ (điền số tài khoản, hệ thống tự động truy vấn các thông tin còn lại, KHÔNG truy vấn số dư, của người nợ), hoặc chọn người nợ trong danh sách đã lưu.
- Nhập số tiền chuyển & nội dung nhắc nợ
- Ra lệnh GỬI NHẮC NỢ
- Nhắc nợ sẽ được gửi đến cho người nợ. Người nợ xem được nhắc nợ này ở màn hình **Xem danh sách nợ chưa thanh toán**

### 1.5.2 Xem danh sách nợ

- Gồm danh sách do bản thân tạo & danh sách nhắc nợ do người khác gửi

### 1.5.3 Huỷ nhắc nợ

- Chọn nhắc nợ cần xoá, nhập nội dung và xác nhận xoá
- Nếu là nhắc nợ do bản thân tạo ➡ gửi notify cho người nợ tương ứng
- Nếu là nhắc nợ do người khác gửi ➡ gửi notify cho người gửi

### 1.5.4 Thanh toán nhắc nợ

- Hệ thống yêu cầu xác nhận OTP và tự động thực hiện 1 giao dịch chuyển khoản nội bộ từ người nợ ➡ người nhắc.
- Nhắc nợ được ghi nhận trạng thái **đã thanh toán**, đồng thời notify cho người nhắc.

## 1.6 Xem lịch sử giao dịch của 1 tài khoản

- Giao dịch nhận tiền
- Giao dịch chuyển khoản
- Giao dịch thanh toán nhắc nợ
- Được sắp xếp từ mới đến cũ, có thể hiện khác nhau cho các loại thanh toán khác nhau

## 1.7 Đổi mật khẩu

- Mật khẩu được mã hoá bằng thuật toán **bcrypt** hoặc **scrypt**

## 1.8 Quên mật khẩu

- Yêu cầu xác nhận bằng email OTP

## 2. Phân hệ giao dịch viên **employee**

---

### 2.1 Tạo tài khoản khách hàng

- Thông tin đăng nhập
- Thông tin cá nhân { họ tên, email, phone }
- Hệ thống tự phát sinh 01 **tài-khoản-thanh-toán** cho tài khoản khách hàng

### 2.2 Nạp tiền vào 1 **tài-khoản** bất kỳ

- Khách cung cấp thông tin tài khoản (tên-đăng-nhập hoặc số-tài-khoản)
- Khách cung cấp số tiền cần nạp
- Hệ thống ghi nhận

### 2.3 Xem lịch sử giao dịch của 1 tài khoản khách hàng

- Giao dịch nhận tiền
- Giao dịch chuyển khoản
- Giao dịch thanh toán nhắc nợ
- Được sắp xếp từ mới đến cũ, có thể hiện khác nhau cho các loại thanh toán khác nhau

## 3. Cung cấp API để liên kết với ngân hàng khác

---

- API truy vấn thông tin tài khoản

- API nộp tiền vào tài khoản

## Yêu cầu bảo mật

- Để đảm bảo tính bảo mật, các ngân hàng đều chọn một cơ chế mã hoá bất đối xứng cho các api của mình.
  - **RSA**. Private key và public key có thể phát sinh ở <https://cryptotools.net/rsagen> (<https://cryptotools.net/rsagen>), hoặc google với từ khoá **rsa key generator**
  - **PGP**. Private key và public key có thể phát sinh ở <https://pgpkeygen.com> (<https://pgpkeygen.com>), hoặc google với từ khoá **pgp key generator**
- Khi ngân hàng B muốn truy cập các api do ngân hàng A cung cấp, quá trình diễn ra như sau
  - A kiểm tra lời gọi api có phải xuất phát từ B (đã đăng ký liên kết từ trước) hay không
  - A kiểm tra xem lời gọi này là mới hay là thông tin cũ đã quá hạn
  - A kiểm tra xem gói tin B gửi qua là gói tin nguyên bản hay gói tin đã bị chỉnh sửa
    - Sử dụng kỹ thuật hash gói tin với secret key quy định trước
  - Nếu là yêu cầu liên quan đến tiền bạc (nạp tiền, trừ tiền), A thực hiện thêm công đoạn **verify** chữ ký bất đối xứng (RSA/PGP) mà B ký vào yêu cầu nạp/trừ tiền, nếu chữ ký hợp lệ, A mới thực hiện api. Response trả cho B cũng được A ký bất đối xứng tương ứng.
    - Chữ ký bất đối xứng là cơ sở để A và B thanh toán tiền bạc cho nhau (đối soát)

## 4. Phân hệ quản trị viên - **administrator**

---

- Quản lý danh sách nhân viên
  - Các chức năng **quản lý** cơ bản
- Xem danh sách giao dịch trong tháng với các ngân hàng khác (đối soát)
  - Xem trong khoảng thời gian
  - Xem theo từng ngân hàng, hoặc tất cả ngân hàng liên kết
  - Có thống kê tổng số tiền đã giao dịch

## 5. Các yêu cầu khác

---

### 5.1 Yêu cầu kỹ thuật

- SPA (Frontend)
  - **reactjs/vuejs/angular/svelte/...**
  - **react-router/vue-router/...**

- **redux/vuex/...**
- Backend
  - RESTful API
  - db: **mysql/postgres/mongodb**
- Chỉ hoàn thành **ĐÚNG** các chức năng được yêu cầu
- Mọi api nội bộ đều phải cài đặt JWT **access-token** và **refresh-token**

## 5.2 Yêu cầu dữ liệu

- Cần ít nhất 8 tài khoản, mỗi tài khoản có ít nhất 8 giao dịch nhận & chuyển tiền
- Cần kết nối với ít nhất 01 hệ thống ngân hàng khác
  - 1 ngân hàng bảo mật PGP (21HCB không cần làm)
  - 1 ngân hàng bảo mật RSA

## 5.3 Yêu cầu quản lý mã nguồn

- Sinh viên cần upload mã nguồn lên **github** từ lúc bắt đầu thực hiện đồ án.
- Nhóm nào lịch sử commit/push gần như không có ➡ 0đ.