

Báo cáo tuần (lần 5)

Mã sinh viên: B22DCCN650

Họ và tên: Nguyễn Việt Quang

Tên đề tài: Online learning website

A. Nội dung đã làm và tìm hiểu trong tuần qua

Tìm hiểu về cách tích hợp thanh toán zalopay

Tìm hiểu về cách lưu video trên cloud của aws

Các chức năng thực hiện được bên front end:

- Đăng kí cho khóa học miễn phí

- Xem các khóa học đã đăng kí

- Xem video của khóa học

Các công việc thực hiện được phần back end:

- Xây dựng api:

 - Tạo hóa đơn

 - Kiểm tra xem khóa học đã đăng kí chưa

Cách tích hợp thanh toán zalopay

Đăng ký tài khoản ZaloPay Merchant

Truy cập: <https://merchant.zalopay.vn>

Đăng ký tài khoản và đăng ký ứng dụng thanh toán

Sau khi duyệt, hệ thống cung cấp:

app_id

key1 (dùng tạo chữ ký)

key2 (dùng kiểm tra trạng thái giao dịch)

Cài đặt backend (Java Spring Boot)

Thêm thư viện HTTP client (nếu chưa có):

```
<!-- pom.xml --><dependency>
  <groupId>com.squareup.okhttp3</groupId>
  <artifactId>okhttp</artifactId>
  <version>4.12.0</version></dependency>
```

Tạo lớp xử lý ZaloPayService.java:

```

public class ZaloPayService {
    private static final String APP_ID = "2554";
    private static final String KEY1 = "your_key1";
    private static final String CREATE_ORDER_URL =
"https://sandbox.zalopay.com.vn/v001/tpe/createorder";

    public String createOrder(String appTransId, long amount, String
description) throws Exception {
        Map<String, Object> order = new HashMap<>();
        order.put("app_id", APP_ID);
        order.put("app_trans_id", appTransId);
        order.put("app_user", "demo_user");
        order.put("app_time", System.currentTimeMillis());
        order.put("amount", amount);
        order.put("item", "");
        order.put("description", description);
        order.put("embed_data", "{}");
        order.put("bank_code", "zalopayapp");

        String data = APP_ID + "|" + appTransId + "|" + order.get("app_user") +
        "|" + amount + "|" + order.get("app_time") + "|" + order.get("embed_data") + "|" +
        order.get("item");
        String mac = hmacSha256(data, KEY1);
        order.put("mac", mac);

        OkHttpClient client = new OkHttpClient();
        RequestBody body = RequestBody.create(MediaType.get("application/x-
www-form-urlencoded"),
            "data=" + URLEncoder.encode(new Gson().toJson(order), "UTF-
8"));

        Request request = new Request.Builder()
            .url(CREATE_ORDER_URL)
            .post(body)
            .build();

        Response response = client.newCall(request).execute();
        return response.body().string();
    }

    private String hmacSha256(String data, String key) throws Exception {
        SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(),
"HmacSHA256");
        Mac mac = Mac.getInstance("HmacSHA256");
        mac.init(secretKeySpec);
        byte[] hash = mac.doFinal(data.getBytes());
        return DatatypeConverter.printHexBinary(hash).toLowerCase();
    }
}

```

Gọi API từ frontend (React hoặc HTML)

Giao diện React (gửi request tạo order):

```
fetch('/api/payment/create', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({ amount: 100000 })
})
.then(res => res.json())
.then(data => {
  window.location.href = data.order_url;
});
```

Nhận callback từ ZaloPay

ZaloPay sẽ gửi POST request tới callback_url bạn cấu hình. Backend cần xác minh chữ ký để đảm bảo tính toàn vẹn:

```
@PostMapping("/payment/callback")public ResponseEntity<String>
handleCallback(@RequestBody Map<String, Object> body) {
    String data = (String) body.get("data");
    String reqMac = (String) body.get("mac");
    String mac = hmacSha256(data, KEY2);

    if (!mac.equals(reqMac)) return
    ResponseEntity.badRequest().body("INVALID MAC");
    // xử lý cập nhật trạng thái đơn hàng ở đây

    return ResponseEntity.ok("{\"return_code\":1}");
}
```

Kiểm tra và triển khai

Kiểm thử trên môi trường **sandbox**

Kiểm tra các case: thanh toán thành công, hủy giao dịch, callback

Chuyển sang **production** sau khi kiểm tra đầy đủ

Cách lưu video lên cloud của aws

Tạo bucket trên Amazon S3

Đăng nhập vào AWS Console: <https://aws.amazon.com>

Tìm dịch vụ **S3**.

Chọn **Create bucket**.

Nhập tên bucket (phải là duy nhất toàn cầu).

Chọn region (ví dụ: ap-southeast-1 – Singapore).

Bỏ chọn **Block all public access** nếu bạn muốn video có thể xem công khai (hoặc để mặc định nếu dùng xác thực).

Nhấn **Create bucket**.

Thiết lập quyền truy cập (IAM)

Truy cập dịch vụ **IAM**.

Tạo user hoặc role với quyền AmazonS3FullAccess hoặc giới hạn quyền truy cập theo bucket.

Tải video lên S3 bằng SDK hoặc giao diện

Dùng AWS SDK (Python example)

Cài đặt:

```
pip install boto3
```

Mã Python:

```
import boto3 from botocore.exceptions import NoCredentialsError
```

```
ACCESS_KEY = 'YOUR_ACCESS_KEY'
SECRET_KEY = 'YOUR_SECRET_KEY'
BUCKET_NAME = 'your-bucket-name'
def upload_video(file_path, object_name):
    s3 = boto3.client('s3',
                      aws_access_key_id=ACCESS_KEY,
                      aws_secret_access_key=SECRET_KEY)

    try:
        s3.upload_file(file_path, BUCKET_NAME, object_name)
        print("Upload thành công")
    except FileNotFoundError:
        print("File không tồn tại")
    except NoCredentialsError:
        print("Lỗi xác thực AWS")
# Ví dụ dùng:
upload_video('video.mp4', 'videos/video.mp4')
```

Tạo URL để truy cập video

Công khai (nếu file public):

