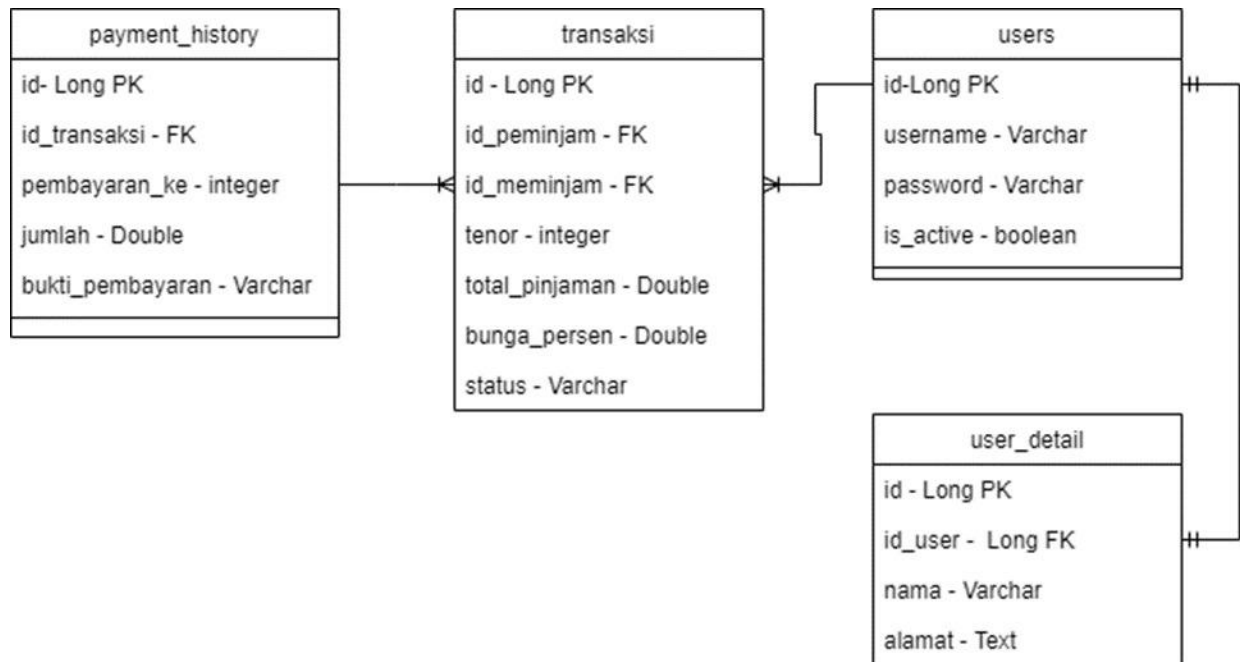


Berikut merupakan deskripsi soal yang perlu anda selesaikan pada tugas akhir ini



1	<p>Buatlah kelas entitas dan repository sesuai ERD pada gambar diatas. Nama kelas, atribut dan type data sesuaikan. Kelas entitas pada package "model". Kelas repository pada package "repository". Masukkan url gitlab anda pada file Template Jawaban dan screenshot file mana saja yang mencakup soal no 1. Nama branch soal-satu</p>
2	<p>Buatlah Interface "TransaksiService" pada package "view", berisikan method sebagai berikut.</p> <ol style="list-style-type: none"> nama method save dengan type method Map dengan parameter kelas "Transaksi" nama method updateStatus dengan type method Map dengan parameter kelas "Transaksi" <p>Masukkan url gitlab anda pada file Template Jawaban dan screenshot file mana saja yang mencakup soal no 2. Nama branch soal-dua</p>
3	<p>Buatlah kelas "TransaksiPaymentImple" pada package "view.impl", dimana mengimplementasi interface "TransaksiService".</p> <ol style="list-style-type: none"> buatlah logic pada method save, supaya dapat menyimpan data pada tabel "Transaksi". buatlah logic pada method updateStatus, yang diupdate hanya atribut "status" dengan value "lunas". <p>Masukkan url gitlab anda pada file Template Jawaban dan screenshot file mana saja yang mencakup soal no 3. Nama branch soal-tiga</p>

4	<p>Buatlah class TransaksiController pada package controller. Request Mapping Berisikan method berikut:</p> <p>a. nama method “save” dengan type ResponseEntity<Map>, memanggil method “save” pada interface “TransaksiService. Type PostMapping dengan Response sebagai berikut: post : localhost:8080/v1/transaksi</p> <pre>{ "data": { "..": "..", "..": ".." }, "status": "sukses", "code": "200" }</pre> <p>b. nama method “updateStatus” dengan type ResponseEntity<Map>, memanggil method “updateStatus” pada interface “TransaksiService. Type PutMapping dengan Response sebagai berikut put : localhost:8080/v1/transaksi</p> <pre>{ "data": { "..": "..", "..": ".." }, "status": "sukses", "code": "200" }</pre> <p>c. nama method “list” dengan type ResponseEntity<Transaksi>, menampilkan semua transaksi dengan pagination. Jika request param diisi, maka akan memfilter berdasarkan status(query menggunakan ‘like’), jika tidak akan menampilkan semua data. Request param sebagai berikut : Request Param Integer page, Request Param Integer size, Request Param String status Response sebagai berikut: get : localhost:8080/v1/transaksi/list</p> <pre>{ "content": [{}], "pageable": {}, "totalPages": 2, "last": false, "totalElements": 2,</pre>
---	---

	<pre> "size": 1, "number": 0, "sort": {}, "first": true, "numberOfElements": 1, "empty": false } </pre> <p>Ketentuan : query findAll harus menggunakan query yang dibuat sendiri pada repository transaksi contoh :</p> <pre> @Query("select c from tabel c") Page<tabel> getList(Pageable pageable); </pre> <p>Masukkan url gitlab anda pada file Template Jawaban dan screenshot file mana saja yang mencakup soal no 4. Nama branch soal-empat</p>
5	<p>Setelah soal 1 dan 4 selesai, running aplikasi anda. Kemudian lakukan testing pada rest API yang telah anda buat dengan menggunakan rest template dan JUNIT.</p> <p>Buatlah class "TestingController" pada package testing. Buatlah method void restTemplateSave, digunakan untuk memanggil rest api POST http://localhost:8081/api/v1/transaksi. Jika sukses maka akan menyimpan pada tabel transaksi di database.</p> <p>Masukkan url gitlab anda pada file Template Jawaban dan screenshot file mana saja yang mencakup soal no 5.</p>

6	<p>Tambahkan Spring Security dengan Oauth JWT token pada REST API yang telah anda kerjakan pada soal diatas.</p> <p>Langkah yang dapat anda lakukan adalah</p> <ul style="list-style-type: none"> - Add dependency di pom.xml - Add file entity di package model - Add file repository - Add file service di package view - Add file di package config <p>Tambahkan pada pom.xml</p> <pre> <dependency> <groupId>org.springframework.security.oauth</groupId> <artifactId>spring-security-oauth2</artifactId> <version>\${oauth2.version}</version> </dependency> <dependency> <groupId>org.springframework.security</groupId> <artifactId>spring-security-jwt</artifactId> <version>\${jwt.version}</version> </dependency> <dependency> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-security</artifactId> </dependency> <dependency> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-data-rest</artifactId> </dependency> </pre> <p>Jika berhasil, outputnya seperti gambar dibawah ini :</p> <p>Silahkan testing di postman</p> <ul style="list-style-type: none"> - POST /v1/transaksi - PUT /v1/transaksi - GET /v1/transaksi/list <p>Masukkan url gitlab anda pada file Template Jawaban dan screenshot file mana saja yang mencakup soal no 6. Output (output login, POST /v1/transaksi,PUT /v1/transaksi,GET /v1/transaksi/list).</p>
---	---