

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Universitas Negeri Jakarta (UNJ) merupakan salah satu Perguruan Tinggi Negeri (PTN). Program pendidikan dilakukan dengan kegiatan pembelajaran terhadap jenjang Diploma, Sarjana, Magister, hingga Doktor. Program tersebut dilakukan dalam menunjang pelayanan pendidikan masyarakat di Indonesia.

Biaya operasional dalam rangka pelaksanaan kegiatan dan pembangunan pada PTN menerapkan sistem Biaya Kuliah Tunggal (BKT) dan Uang Kuliah Tunggal (UKT) yang ditetapkan berdasarkan Peraturan Menteri Riset, Teknologi, dan Pendidikan Tinggi (PERMENRISTEKDIKTI) Nomor 39 Tahun 2017 yang berisi bahwa BKT merupakan keseluruhan biaya operasional mahasiswa per semester, sedangkan UKT merupakan biaya yang akan ditanggung atau dibayarkan setiap mahasiswa. UKT merupakan sebagian dari dana masyarakat yang diatur oleh pemerintah untuk memenuhi kebutuhan pendidikan yang di Indonesia.

UKT terdiri atas beberapa kelompok yang diusulkan oleh pihak PTN untuk ditetapkan oleh menteri RISTEKDIKTI. UKT dikelompokan berdasarkan kemampuan ekonomi mahasiswa atau orang tua/wali yang membiayainya. Pimpinan PTN dapat memberlakukan UKT yang sesuai dengan kondisi ekonomi mahasiswa. UKT tersebut selanjutnya dibayarkan oleh mahasiswa sesuai dengan media pembayaran yang telah ditentukan oleh PTN terkait seperti bank.

Transaksi pembayaran di UNJ, menggunakan jasa bank sebagai media pembayaran. Proses pembayaran melayani pembayaran seperti UKT, Sumbangan Pembangunan Universitas (SPU), biaya administrasi cuti, pendaftaran Penerimaan

Mahasiswa Baru (PENMABA), dan lain-lain. UNJ bekerja sama dengan empat bank yaitu Bank Mandiri, Bank Negara Indonesia (BNI), Bank Tabungan Negara (BTN), dan Bank Bukopin.

Meskipun sudah bekerja sama dengan empat bank, namun sistem pembayaran tidak bersifat *Multibank*. Masing-masing bank secara terpisah menangani jenis pembayaran dan kelompok mahasiswa yang berbeda-beda. Hal ini dapat menimbulkan permasalahan dan kesulitan pada bagian-bagian tertentu. Seperti pembayaran yang kurang fleksibel, serta kinerja dalam hal laporan keuangan. Permasalahan lain yang dapat timbul yaitu penanganan terhadap sistem yang terdapat pada internal bank. Permasalahan internal dari bank, tidak dapat dikontrol secara menyeluruh dan akan menyulitkan pembayaran pada jenjang-jenjang tertentu.

Sistem pembayaran di UNJ menerapkan sistem yang dinamakan *Host to Host* (H2H). H2H merupakan sebuah solusi otomatis untuk melakukan pembayaran dengan media transfer data elektronik yang aman antara bank dengan klien korporat. H2H memiliki beberapa kelebihan seperti *Seamless, Security, Multi Transaction, Currency Support*, dan lain-lain.

Dikutip dari beberapa aplikasi *web* UNJ, kelompok pembayaran di UNJ saat ini terbagi menjadi beberapa kelompok. Pembayaran jenjang Strata-1 (S1) untuk angkatan 2016 ke bawah, dibayarkan melalui sistem H2H BNI. Seluruh angkatan jenjang Diploma-3, dan seluruh jenjang S1 pada angkatan 2016 ke atas yakni: 1) Fakultas Ilmu Sosial (FIS); 2) Fakultas Teknik; dan 4) Fakultas Ekonomi, menggunakan sistem H2H Bank Mandiri. Untuk seluruh jenjang S1 pada angkatan 2016 ke atas yakni: 1) Fakultas Ilmu Olahraga (FIO); 2) Fakultas Bahasa dan Seni

(FBS); 3) Fakultas Matematika; 4) Ilmu Pengetahuan Alam (FMIPA); 5) Fakultas Ilmu Pendidikan (FIP); dan 6) Fakultas Pendidikan Psikologi (FPPsi), menggunakan sistem H2H BTN. Sedangkan pembayaran untuk jenjang Pascasarjana, menggunakan H2H Bank Bukopin. Setiap sistem H2H tersebut akan dikembangkan dan diteliti untuk pengembangan sistem pembayaran *Multibank*. Sistem pembayaran ini dikembangkan oleh Unit Pelaksana Teknis Teknologi Informasi dan Komunikasi (UPT TIK) yang ada di UNJ.

Untuk meningkatkan layanan agar dapat menerapkan sistem pembayaran yang fleksibel dan mudah untuk dikontrol, UPT TIK UNJ berencana mengembangkan sistem *Multibank*. Pembayaran yang fleksibel dapat membuat mahasiswa melakukan pembayaran tanpa adanya penggolongan seperti pada fakultas-fakultas tertentu dan keterikatan membayar hanya pada bank-bank yang telah ditetapkan. Pembayaran dapat dilakukan di semua bank yang bekerja sama dengan UNJ dan mengimplementasikan sistem *Multibank* ini.

Pengembangan *Multibank* akan menggunakan teknologi *web* dalam bertukar informasi tentang detail tagihan dan status pembayaran. Pengembangan akan mengimplementasikan *website* yang bersifat *backend* (bagian belakang layar atau *server*). Teknologi tersebut dapat memenuhi kebutuhan fungsional yang ada pada dokumen *Software Requirement Specification* (SRS).

Pengembangan *web* pada sisi *backend* memprioritaskan modul pemrosesan data, serta *business logic* pada instansi tersebut. Penggunaan *backend* sering kali dikaitkan dengan istilah *web service*, penyedia layanan data melalui akses dari sebuah alamat *web*. Selain *backend*, terdapat *frontend* yang merupakan pengembangan *web* yang dibentuk sebagai tampilan yang berada pada sisi

pengguna. Gabungan antara *backend* dan *frontend* dapat menjadi sebuah aplikasi *web*, serta pengembangan yang cepat, efisien dan bersifat modular/atomik (pengembangan per komponen aplikasi).

*Web Service* merupakan sebuah aplikasi yang terletak pada sisi *server*. *Web Service* dapat diakses melalui sebuah Uniform Resource Locator (URL) atau alamat *web*, seperti alamat yang ditemukan pada antarmuka aplikasi *web browser*. Setiap URL dari *web service* merupakan bagian daripada Uniform Resource Identifier (URI). Setiap URL yang diakses melayani proses tertentu, kemudian menampilkan sebuah respons atau tampilan dari permintaan yang diberikan oleh pengakses URL. Setiap respons tersebut memiliki berbagai bentuk format (berupa data atau sebuah tampilan) yang berbeda satu sama lain. Sebelum memberikan respons, *web service* seringkali melakukan proses *business logic* pada setiap layanannya. Proses tersebut berupa manipulasi data dan/atau tampilan sebagai bentuk dari jawaban yang akan diberikan.

Pengembangan *Web Service* akan membentuk sebuah aplikasi *backend* dengan implementasi arsitektur *Service Oriented Architecture* (SOA) dan *data interchange format* JSON. Arsitektur atau gaya penulisan SOA pada sebuah *web service* merupakan sekumpulan dari beberapa alamat terakhir yang disebut sebagai *endpoint* dari sebuah URL. Berbagai *endpoint* menghasilkan keluaran atau respon yang khusus dan berbeda. Dikutip dari <https://www.opengroup.org/soa/source-book/soa/p4.htm>, SOA memiliki keuntungan seperti: 1) Kemampuan untuk mengembangkan kombinasi fungsi baru dengan cepat; 2) Biaya pengembangan dana manajemen yang rendah; dan 3) Kemampuan untuk beradaptasi ke dalam lingkungan yang baru yang terhitung cepat. Data yang dikeluarkan dari setiap

*endpoint* akan diformat ke dalam bentuk JSON. Dibandingkan dengan format XML, JSON memiliki keunggulan dalam hal kecepatan pengiriman dan pemrosesan data. JSON juga dapat diinterpretasikan ke berbagai bahasa pemrograman seiring dengan perkembangan teknologi ([https://www.cs.tufts.edu/comp/150IDS/final\\_papers/tstras01.1/FinalReport/FinalReport.html](https://www.cs.tufts.edu/comp/150IDS/final_papers/tstras01.1/FinalReport/FinalReport.html)). Arsitektur SOA dan format JSON, dapat memenuhi kebutuhan yang ada pada dokumen SRS. SOA dan JSON juga memudahkan pendokumentasian dan pengembangan terhadap sistem. Selain arsitektur, *web service* juga akan mengimplementasikan bahasa pemrograman JavaScript. Dengan bantuan teknologi Node.js, aplikasi bahasa JavaScript dapat bekerja diluar *browser*, sehingga pengembangan *web service* dengan JavaScript tidak menjadi mustahil. *Web Service* yang dikembangkan, menggunakan salah satu *framework* JavaScript yaitu Express dalam menerapkan arsitektur SOA secara teknisnya.

Pengembangan *web service* pada sistem *Multibank* diharapkan dapat menghasilkan sejumlah aplikasi yang lebih mudah untuk dikembangkan. Mahasiswa dari berbagai jenjang dapat membayar tagihan pembayarannya pada bank dan kanal apapun. Selain pelayanan, rekapitulasi keuangan serta pengembangan sistem menjadi lebih mudah karena format yang tersusun dan menyatu. Data tagihan serta pembayaran yang dihasilkan juga akan memiliki format yang konsisten pada setiap banknya.

Berdasarkan permasalahan tersebut, hal ini menguatkan peneliti perlu adanya *Web Service Multibank* yang dapat menangani serta memproses pembayaran secara terpadu dan *database* yang terdokumentasi dengan baik.

## 1.2. Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan, dapat diidentifikasi permasalahan-permasalahan yang ada pada berikut ini:

1. Meskipun sudah bekerja sama dengan empat bank, kebijakan yang ditetapkan untuk setiap mahasiswa yaitu hanya dapat membayar pada bank yang telah ditentukan;
2. Masing-masing H2H memiliki perbedaan struktur *database*, sehingga menyulitkan pada saat melacak dan mengambil data;
3. Belum adanya penerapan sistem *Multibank* di Universitas Negeri Jakarta

## 1.3. Batasan Masalah

Melihat luas ruang lingkup dari permasalahan yang diidentifikasi sebelumnya, batasan masalah yang dapat diidentifikasi adalah:

1. Penelitian menelaraskan struktur *database* dan menyatukan setiap H2H menjadi sebuah *web service*;
2. Fokus penelitian hanya pada pengembangan *web service*, tidak menghasilkan aplikasi yang lengkap.

## 1.4. Rumusan Masalah

Berdasarkan latar belakang, identifikasi masalah, dan pembatasan masalah, maka perumusan masalah yang akan dibahas pada penelitian ini adalah “Bagaimana Mengembangkan *Web Service* untuk sistem pembayaran *Multibank* di Universitas Negeri Jakarta?”

### **1.5. Tujuan Penelitian**

Berdasarkan rumusan masalah pada pembahasan sebelumnya, tujuan dari penelitian ini adalah:

1. Menghasilkan *web service* dan *database* transaksi pembayaran bank yang bekerja sama dengan instansi terkait.

### **1.6. Manfaat Penelitian**

Manfaat dari penelitian ini dapat dibagi menjadi dua bagian, yaitu manfaat teoritis dan manfaat praktis.

1. Secara teoritis, hasil penelitian ini diharapkan dapat menjadi sumber kajian dan bahan referensi penelitian lainnya, terkait dengan bidang ilmu pengembangan *database*, *web service* dan teknologi *web* yang lain;
2. Secara praktis, hasil aplikasi penelitian ini dapat digunakan dan disinkronkan dengan aplikasi *frontend*, sehingga membentuk sebuah aplikasi yang lengkap.