

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi kini semakin pesat hingga digunakan di seluruh bidang kehidupan. Hal ini berbanding lurus dengan meningkatnya kebutuhan kompetensi generasi muda agar relevan dengan perkembangan zaman. Hal ini didukung dengan perubahan pada bidang pendidikan, dimana sekarang dapat menimba ilmu dimana saja dan kapan saja. Mudah didapatkan sumber belajar *online* yang memberikan fleksibilitas dalam belajar, materi terbaru, dan dibuat oleh orang yang berpengalaman di bidangnya. Fenomena ini sering dikenal dengan ledakan *non degree*, dimana orang yang tidak memiliki gelar pendidikan tinggi bisa saja lebih baik dari pada yang memilikinya [1].

Oleh karena itu, meningkatkan kompetensi mahasiswa baik *soft skills*, *hard skills*, dan pengalaman yang relevan dengan kompetensi industri saat ini menjadi prioritas utama agar mahasiswa siap dan dapat bersaing di dunia industri. Mahasiswa memerlukan informasi dan pengalaman tentang standar kerja yang profesional pada jenjang karier yang akan ditempuh oleh mahasiswa. Dan juga jaringan kerja sangat berpengaruh pada masa depan calon pekerja, oleh karena itu mahasiswa perlu mengembangkan jaringan kontak profesional untuk referensi di masa depan. Semua ini bisa didapatkan melalui program kerja magang.

Selanjutnya dengan perkembangan teknologi pada bidang keuangan, sekarang mulai beralih pada sistem automasi menggunakan teknologi yang canggih untuk mempermudah pekerjaan dan mengurangi *human error*. Sistem teknologi pada bidang keuangan sering disebut dengan *financial technology* atau *fintech*. Sistem ini dibutuhkan kemampuan untuk *scalability*, memiliki akurasi dan validitas yang tinggi, tahan untuk berbagai kondisi, dan keamanan yang tinggi. Dimana sistem ini didesain untuk melayani transaksi pengguna dengan jumlah besar sehingga dibutuhkan sistem teknologi keuangan yang mumpuni dan optimal sehingga memenuhi *business needs* suatu perusahaan.

Salah satu perusahaan yang menggunakan sistem keuangan secara otomasi adalah PT Traveloka Indonesia. Traveloka merupakan perusahaan berbasis

teknologi yang menawarkan jasa dan berfokus pada pembelian tiket transportasi dan juga hotel secara *online* untuk memenuhi kebutuhan *lifestyle* penggunanya. Traveloka yang berdiri sejak tahun 2012 sekarang memiliki jutaan pengguna yang aktif setiap bulannya dan merupakan salah satu perusahaan teknologi terbesar dan terbaik se-Asia Tenggara.

Salah satunya, Traveloka menggunakan sistem *payment out* untuk mentransfer uang ke berbagai supplier dan vendor Traveloka seperti hotel, *accommodation*, transportasi, dan sebagainya. Namun sistem ini belum sepenuhnya optimal dan perlu dilakukan pengembangan lebih lanjut. Ada beberapa faktor penting dalam suatu automasi sistem keuangan, yaitu pentingnya keamanan data, analisis transaksi data valid, dan juga legalitas sistem yang sesuai. Hal ini menjadi salah satu latar belakang proyek yang dikerjakan oleh penulis selama melakukan program kerja magang.

Terutama program kerja magang yang ditawarkan Traveloka ke mahasiswa adalah mahasiswa akan mendapatkan kesempatan dan pengalaman untuk bekerja dan berkontribusi menyelesaikan proyek yang relevan dengan konteks bisnis zaman sekarang (*real-world problem*), berkolaborasi dan berkoordinasi dengan talenta Traveloka dari berbagai latar belakang dan berpengalaman di bidangnya, dan mengeksplorasi pengembangan diri dengan ilmu baru sesuai jalur karier yang diminati dan preferensi pribadi. Hal ini sesuai dengan prioritas utama dalam mengembangkan kompetensi mahasiswa agar relevan dengan dunia industri sekarang.

Oleh latar belakang tersebut laporan ini berisi tentang penulis melaksanakan program kerja magang pada PT Traveloka Indonesia sebagai *backend engineer intern* di divisi *Payment Out*. Penulis ikut berkontribusi dan berkolaborasi pada pembuatan proyek nyata fitur fitur baru untuk mengoptimisasi dan migrasi sistem *Payment Out* Traveloka sehingga meningkatkan kualitas sistem secara keseluruhan. Penulis mengembangkan fitur *tooling*, *reporting*, *tracking*, sentralisasi penggunaan data *microservices*, dan pembaharuan sistem transfer bank.

1.2 Maksud dan Tujuan Pelaksanaan Kerja Magang

Dengan mengikuti program kerja magang di PT Traveloka Indonesia, penulis memiliki tujuan untuk mendapatkan kompetensi dan pengalaman yang relevan dengan dunia industri profesional zaman ini. Penulis akan terjun langsung mengerjakan proyek nyata yang akan membawa dampak terhadap bisnis Traveloka dengan didampingi oleh mentor yang berpengalaman.

Selama melaksanakan kerja magang, penulis memiliki maksud dan tujuan sebagai berikut:

1. Mendapatkan kesempatan untuk bekerja dan berkontribusi secara nyata dengan menciptakan dampak dan menyelesaikan proyek yang relevan dengan konteks bisnis zaman sekarang (*real-world problem*) pada salah satu perusahaan teknologi terbesar di Asia Tenggara.
2. Mendapatkan kesempatan belajar dengan talenta dan mentor terbaik dari berbagai latar belakang universitas lokal maupun internasional dan memiliki banyak pengalaman di bidangnya.
3. Mempelajari dalam proses pembuatan program *backend* untuk sistem automasi keuangan secara *best practice* dan standar industri dengan alur kerja profesional.

Proyek program kerja magang yang dikerjakan oleh penulis memiliki tujuan untuk mengembangkan *backend* pada sistem *payment out* di PT Traveloka Indonesia demi meningkatkan kualitas sistem secara keseluruhan. Penulis diberikan kepercayaan dan kesempatan untuk bertanggung jawab pada berbagai proyek yang krusial pada sistem *payment out* seperti *tooling*, *reporting*, *tracking*, migrasi sistem transfer, dan sentralisasi penggunaan data *microservices*. Di antara proyek proyek tersebut terdapat maksud dan tujuan, yaitu :

1. Meningkatkan kualitas operasional sistem dengan mengurangi akses langsung *database* melalui pembuatan fitur *tooling* untuk menghindari risiko operasi perubahan data yang penting.
2. Meningkatkan validitas sistem dengan menyimpan dan melaporkan data yang valid dan sesuai dengan keperluan perusahaan.

3. Menyesuaikan legalitas sistem dengan melakukan migrasi API transfer ke versi yang lebih baru dan berstandar nasional.
4. Mengirimkan data yang valid untuk setiap transaksi di sistem *payment out* kepada tim data sehingga sistem dapat dianalisis
5. Mengurangi duplikasi data dan kesalahan operasional dengan melakukan sentralisasi penggunaan data pada beberapa *microservice*.

1.3 Waktu dan Prosedur Pelaksanaan Kerja Magang

1.3.1 Waktu Pelaksanaan Kerja Magang

Pelaksanaan program kerja magang berlangsung dari tanggal 16 Agustus 2022 sampai dengan 31 Desember 2022 selama 20 minggu. Jam kerja magang pada hari kerja dimulai pada jam 9.00 sampai dengan jam 17.00 WIB dengan durasi 8 (delapan) jam kerja dan 1 (satu) jam istirahat pada jam 12.00 sampai dengan jam 13.00 WIB.

Rangkaian dasar pelaksanaan program kerja diuraikan dengan kerangka sebagai berikut:

Tabel 1.1 Tabel Rangkaian Kerangka Program Kerja magang Traveloka

Rangkaian Kegiatan	Tujuan	Durasi
<i>Onboarding</i>	Penulis mendapatkan materi pengenalan dari Traveloka dan terdapat <i>bootcamp</i> untuk mengenalkan <i>code convention</i> yang dipakai. Penulis mempersiapkan aset yang akan dipakai saat proses pengembangan proyek.	2 hari (16 jam)
<i>Immersion Phase</i>	Penulis beradaptasi dengan cara dan budaya kerja di Traveloka dibantu oleh tim dan mentor.	1 minggu (40 jam)

<i>Creation Phase</i>	Penulis mengerjakan proyek berdasarkan kasus nyata (real-world problem) dengan didampingi oleh mentor dan terdapat <i>checkpoint</i> berkala.	16 minggu (640 jam)
<i>Evaluation</i>	Penulis melakukan presentasi atas proyek yang sudah dikerjakan, berisi konteks proyek, permasalahan dan solusi yang dihadapi penulis, dan membagikan apa yang telah dipelajari oleh penulis.	2 minggu (80 jam)
<i>Personal Mentoring</i>	Penulis berkonsultasi dan melaporkan hasil kerja dengan mentor.	(13 jam)

Pelaksanaan program kerja magang dilakukan secara *hybrid* yaitu bekerja dari rumah dan bekerja dari kantor dengan pembagian 3 hari kerja wajib melalui kantor dan selebihnya dibebaskan bekerja dari mana saja termasuk dapat juga berada di kantor. Selama pelaksanaannya tetap memperhatikan faktor kesehatan seperti mematuhi protokol Kesehatan COVID-19 dengan menggunakan masker dan menjaga jarak.

1.3.2 Prosedur Pelaksanaan Kerja Magang

Pembagian tugas dan alur kerja menggunakan metode Scrum yang direncanakan selama 2 minggu. Setiap *engineer* akan mendapat tugas yang sesuai dengan *bandwith* kerja yang dihitung dengan *story point* setiap pekerjaannya dalam setiap sprintnya, setiap *story point* dikonversikan menjadi 1 (satu) hari kerja. Setiap detail tugasnya akan didokumentasikan melalui aplikasi Jira. Dengan ini manajer dan mentor dapat memonitor pekerjaan setiap penulis. Tugas dapat berupa *assessment*, *development*, *testing*, *deployment*, pembuatan dokumentasi, dan sebagainya.

Pekerjaan yang diberikan adalah hasil dari *sprint planning* dari kebutuhan internal seperti permintaan fitur baru dari tim lain dan juga inisiatif target untuk kuartal 4. Setiap pekerjaan akan didiskusikan terlebih dahulu dengan manajer proyek dan *manager engineer* untuk mendapatkan konteks dan target yang sesuai.

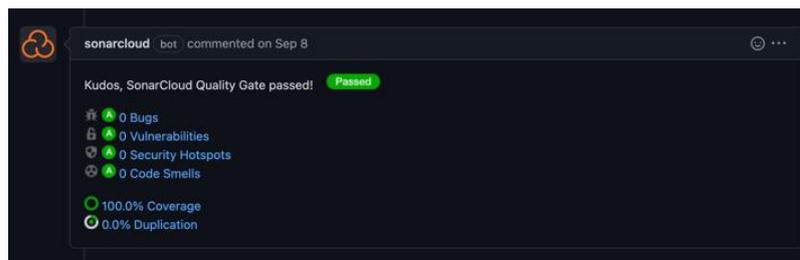
Dalam pengerjaan proyek kerja magang, penulis menggunakan aplikasi IntelliJ IDEA untuk pengembangan *software* Java 11 dan *framework* spring boot, Postman untuk mengirimkan komunikasi HTTP API dan RPC, Github CLI untuk *push* dan *pull* hasil kerja ke repositori dan membuat *pull request*, dokumentasi menggunakan Draw.IO, mermaid JS, dan Swagger, dan berbagai layanan AWS seperti Cloudwatch, EC2, ECS, Parameter Store, dan S3. Dalam pengerjaan penulis dipinjamkan aset perusahaan yaitu Macbook Pro M1 dengan arsitektur ARM.

Untuk kolaborasi proyek dan deployment menggunakan versioning Github dan AWS CodePipeline. pada setiap proyek yang dikembangkan, penulis melakukan *forking branch* master pada repositori utama domain proyek tersebut berada. Ketika proyek yang sudah selesai dikembangkan, penulis membuat *pull request* ke repositori proyek pada Github. Pada *pull request* yang sudah dibuat, SonarLint dan AWS codebuild akan menjalankan proses verifikasi kode untuk mengecek *codesmell*, *bugs*, *duplications*, *code coverage*, dan apakah program berhasil untuk dijalankan. Lalu engineer lain akan *me-review pull request* tersebut, mendiskusikan code, dan nantinya akan mensetujui *pull request* tersebut sehingga perubahan kode dapat di-*merge* ke *branch* master dan akan dilakukan release terjadwal. Penulis juga berkesempatan untuk bertanggung jawab atas lingkup staging dan hotfix application.

Saat dilakukan proses *review pull request* hasil pekerjaan penulis, menurut penulis mentor dan tim *engineer* meninjau dengan sangat teliti untuk mendapatkan kode yang bersih dan sesuai. Dilakukan pengecekan semua konvensi kode yang dipakai di Traveloka seperti *defensive coding*, optimalisasi *looping*, kesesuaian terhadap *business process*, konvensi penamaan, desain arsitektur program, dan sampai validitas *unit test* yang dibuat. Selama melakukan kerja magang, penulis juga sering berdiskusi dengan mentor dan tim domain terkait untuk membahas hasil program yang sudah dibuat di portal Github. Sejauh penulis melakukan kerja

magang belum pernah mendapatkan penolakan *pull request* namun pernah mendapatkan saran dari *reviewer* untuk melakukan perubahan penamaan suatu servis atau model yang lebih sesuai dengan fungsinya agar tidak menimbulkan kebingungan dikemudian hari. Penolakan *pull request* dapat dihindari dikarenakan sebelum mengembangkan program, penulis dan tim mendiskusikan terlebih dahulu perencanaan pengembangan program sehingga menghindari kesalahan pembuatan program.

Setiap servis yang dibuat diharuskan memiliki *coverage* kode *unit test* sebesar 90% dan mengikuti konvensi kode Traveloka seperti indentasi, penamaan variabel, jumlah huruf, dan format camel case. Dimana akan dideteksi sebagai code smell jika tidak mengikuti konvensi tersebut.



Gambar 1.1 Tampilan SonarLint

Tim Traveloka sangat terbuka pada suatu inisiatif dan ide yang diberikan oleh seluruh anggota tim. Pada saat melakukan kerja magang, terdapat beberapa inisiatif yang sudah penulis lakukan seperti optimalisasi suatu *system flow*, pembuatan *function helper*, desain arsitektur program, *API contract*, bagaimana menangani *error*, mengangkat diskusi pada suatu *bug* atau masalah, dan lain lain. Semua inisiatif yang diberikan oleh penulis diterima dengan baik oleh tim. Namun jika pada perubahan yang dilakukan penulis dan anggota tim lain mengetahui perubahan yang lebih baik seperti *library* dan algoritma yang lebih optimal maka penulis akan berdiskusi dengan tim untuk mencari solusi yang terbaik.