

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perguruan tinggi memiliki peran penting dalam melaksanakan Tri Dharma Perguruan Tinggi yang mencakup pendidikan, penelitian, dan pengabdian kepada masyarakat. Dalam pelaksanaan kegiatan tersebut, terdapat kewajiban pajak yang harus dipenuhi oleh perguruan tinggi diantaranya Pajak Penghasilan (PPh) dan Pajak Pertambahan Nilai (PPN). Pemahaman yang baik mengenai kewajiban pajak ini sangat penting untuk memastikan bahwa perguruan tinggi mematuhi peraturan perpajakan yang berlaku [1]

Dalam proses pengelolaan laporan pajak, Direktorat Perencanaan dan Keuangan Universitas Amikom masih bergantung pada penggunaan *google spreadsheet*. Seperti *mengcollect*, menghitung, membayar dan melaporkan pajak penghasilan. Penghasilan pajak berdasarkan Tri Dharma diantaranya honorarium bidang pengajaran, honorarium bidang penelitian dan Abdimas. Namun penggunaan *google spreadsheet* dalam pelaporan pajak menimbulkan ketidakefisienan yang berdampak pada laporan kegiatan pajak.

Oleh karena itu, Direktorat Perencanaan dan Keuangan mempunyai solusi untuk mengembangkan sistem aplikasi berbasis web. Dengan memberikan peluang kepada mahasiswa untuk dapat berkontribusi dalam pengembangan aplikasi pengelolaan pajak melalui prosedur magang. Aplikasi yang akan dikembangkan bernama *One Collecting Agent (OCA)*, aplikasi akan dikembangkan dengan fitur pengintegrasian dan otomatisasi proses pengumpulan, pengelolaan, dan pelaporan perpajakan.

Pengembangan aplikasi akan difokuskan pada pelaporan pajak bagian *agent*. Dengan adanya aplikasi pengelolaan pajak berbasis *website*, diharapkan dapat membantu pihak Direktorat Perencanaan dan Keuangan dalam proses pengelolaan laporan pajak. Serta dapat mempermudah pada saat monitoring dan manajemen laporan kegiatan pajak di setiap proses yang dilakukan oleh Satuan Kerja.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan sebelumnya, maka rumusan masalah yang diangkat yaitu Bagaimana pengembangan dan implementasi Aplikasi *One Collecting Agent (OCA)* yang sesuai dengan kebutuhan user.

1.3 Batasan Masalah

Batasan masalah pada pengembangan produk *One Collecting Agent (OCA)* adalah sebagai berikut:

1. Permasalahan yang diteliti mencakup proses pengumpulan penghasilan tidak teratur dan pelaporan pajak penghasilan atas entitas pemotong Yayasan Amikom Yogyakarta.
2. Batas pengembangan aplikasi meliputi layanan agen yang terdiri dari registrasi wajib pajak, olah penghasilan terkait PPh21, PPh23, PPh4(2) dan inventarisasi PPN, PPh23, PPh4(2) terhadap wajib pajak Yayasan Amikom Yogyakarta.
3. Bahasa pemrograman yang digunakan adalah *Typescript* dan *Node JS*, aplikasi yang dibangun berbasis website
4. Metode pengembangan diterapkan *Rapid Application Development (RAD)* dengan batasan kontribusi pada tahapan pengembangan dan implementasi.

1.4 Tujuan

Tujuan dari penelitian adalah menghasilkan sebuah produk berupa aplikasi *website* yang dapat membantu dalam proses pengumpulan dan pelaporan pajak penghasilan.

1.5 Profil

1.5.1 Profil Mitra Magang IT

Logo Direktorat Perencanaan dan Keuangan, memberikan gambaran visi, misi, dan nilai-nilai yang dijunjung tinggi oleh instansi. Gaya yang ditampilkan cenderung ke bentuk stabilitas yang terdapat unsur spiritual dan dalam segi warna menunjukkan sebuah *sociability*, kepercayaan, dan kejujuran dan inovasi.



Gambar 1. 1 Logo Direktorat Perencanaan dan Keuangan

Direktorat Perencanaan dan Keuangan merupakan direktorat di bawah pimpinan Wakil Rektor 2 bidang Perencanaan, Administrasi dan Keuangan yang bertanggung jawab dalam hal perencanaan dan pengelolaan keuangan Universitas AMIKOM Yogyakarta yang memiliki visi dan misi sebagai berikut:

Visi

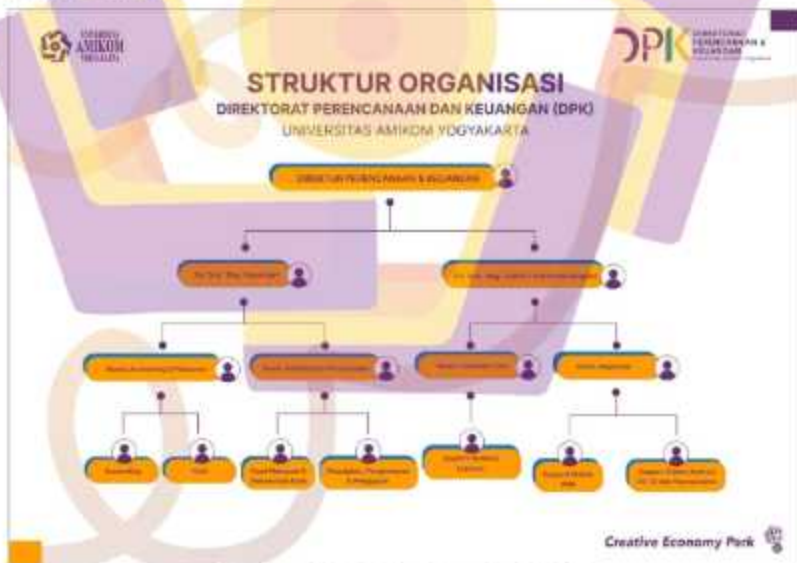
1. Menjadi bagian keuangan yang menjunjung tinggi nilai-nilai kejujuran untuk berperan aktif dalam mendorong terwujudnya tata kelola keuangan UNIVERSITAS AMIKOM Yogyakarta yang akuntabel.
2. Menjadi Bagian Keuangan yang akuntabilitas dan transparansi keuangan serta menuju peningkatan kualitas pelayanan pembayaran mahasiswa di UNIVERSITAS AMIKOM Yogyakarta.
3. Menjadi Bagian Keuangan yang akuntabilitas terhadap sistem transaksi, sistem pembukuan dan pelaporan.
4. Peningkatan kualitas meliputi pelayanan yang cepat, tepat dan memuaskan.
5. Melaksanakan dan meningkatkan mutu pelayanan pembayaran mahasiswa.
6. Melaksanakan tugas pengelolaan keuangan secara efektif dan efisien demi terciptanya tata kelola dan manajemen UNIVERSITAS AMIKOM Yogyakarta yang efektif.
7. Melaksanakan urusan dan layanan administrasi bidang keuangan, meliputi pengelolaan keuangan baik yang bersumber dari Anggaran Pendapatan dan Belanja dari internal UNIVERSITAS AMIKOM Yogyakarta dan eksternal (maupun sumber lain).

Misi

1. Melaksanakan dan meningkatkan mutu pelayanan pembayaran mahasiswa.
2. Melaksanakan tugas pengelolaan keuangan secara efektif dan efisien demi terciptanya tata kelola dan manajemen UNIVERSITAS AMIKOM Yogyakarta yang efektif.
3. Melaksanakan Urusan dan layanan administrasi bidang keuangan, meliputi pengelolaan keuangan baik yang bersumber dari Anggaran Pendapatan dan Belanja dari internal UNIVERSITAS AMIKOM Yogyakarta dan eksternal (maupun sumber lain).

1.5.2 Struktur Organisasi

Direktorat Perencanaan dan Keuangan Universitas Amikom memiliki struktur organisasi yang dipimpin oleh seorang direktur dengan dibantu kasub yang membawahi coordinator dan staff disetiap divisi seperti yang dijabarkan pada gambar berikut:



Gambar 1. 2 Struktur Organisasi DPK

1.5.3 Deskripsi Magang IT

1. Bidang Magang

Dalam pelaksanaan magang yang dilakukan bidang yang diambil adalah pengembangan web dan aplikasi, di mana magang dilakukan dengan tim pengembang untuk merancang, mengkode, menguji, dan meluncurkan aplikasi *website*. Dengan deskripsi *job desk* sebagai berikut:

- a. Berkolaborasi dengan *system analyst* untuk menganalisis kebutuhan sistem dengan melakukan riset kepada *user* atau pengguna.
- b. Berkolaborasi dengan *UI/UX Designer* terkait implementasi desain ke dalam program.
- c. Merancang *database* dari data yang sudah dikumpulkan melalui riset dan menyesuaikannya dengan tampilan desain *UI/UX*.
- d. Membuat *business logic* untuk memastikan bahwa sistem bisa berjalan sesuai dengan kebutuhan *user*.
- e. Melakukan pengujian sistem untuk memastikan aplikasi berjalan dengan baik sesuai dengan kebutuhan *user*.

2. Lokasi Kegiatan

Lokasi kegiatan magang Direktorat Perencanaan dan Keuangan berada di Gedung 4 Lantai 1 Kampus terpadu Universitas Amikom Yogyakarta.

3. Skema Kegiatan

Skema kegiatan magang memberikan gambaran mengenai rencana dan aktivitas yang akan dilakukan selama proses magang berlangsung. Berikut adalah skema magang yang dilakukan pada saat pengembangan aplikasi OCA:



Gambar 1.3 Skema Kegiatan Magang

Kegiatan magang direncanakan berlangsung selama kurang lebih 4 bulan atau 13 minggu, dengan skema kegiatan secara *offline* dan *online*. Pada kegiatan yang akan dilakukan pemberian arahan dan laporan proyek akan dilakukan secara *offline* dan untuk pengerjaan proyek dikerjakan secara *online*. Berikut merupakan deskripsi skema magang yang akan dilakukan.

1. **Minggu 1** > Orientasi dan Pengenalan
Pengenalan tempat instansi magang, struktur organisasi, pengenalan tim magang dan tim IT.
2. **Minggu 2** > Observasi dan Pembelajaran
Mempelajari terkait teknologi, teknik, dan metode serta mempelajari dokumen-dokumen yang berkaitan dengan pembuatan *database* dan kebutuhan pengguna *user* untuk aplikasi yang akan dikembangkan.
3. **Minggu 3 - 4** Pengerjaan Proyek (Analisis)
Menganalisis mengenai perencanaan proyek, analisis kebutuhan, membuat desain sistem, rancangan *database*, yang kemudian dilanjut dengan desain tampilan.
4. **Minggu 5 - 12** Pengerjaan Proyek (Desain, Pengembangan, dan Testing)
Membuat desain tampilan, yang kemudian dikembangkan ke dalam *code*, dan selanjutnya dilakukan pengujian sistem.
5. **Minggu 13** Penyelesaian dan Penilaian
Presentasi proyek magang kepada instansi, dan pemberian nilai dari mentor kepada anggota magang.

6. Minggu 14 Penutupan

Ringkasan kegiatan magang dan penyerahan aplikasi ke instansi magang.

4. Durasi Kegiatan

Pada pelaksanaan kegiatan magang, durasi kegiatan berlangsung kurang lebih selama 6 bulan atau 1 semester. Berikut merupakan penjelasan mengenai kegiatan apa saja yang dilakukan selama proses magang dan durasi yang diperlukan:

1. Orientasi dan Pengenalan 1 Minggu

- Pengenalan instansi magang: Struktur organisasi, visi, misi, dan bidang-bidang pada instansi magang.
- Pengenalan tim: Memperkenalkan tim dan anggota magang, serta tim IT.
- Pengenalan tim IT dan menentukan *development stack* pada sistem yang akan dikembangkan.
- Penjelasan singkat terkait proyek yang akan dikerjakan selama magang.

2. Observasi dan Pembelajaran 3 Minggu

- Melakukan pemilihan beberapa metode pengumpulan kebutuhan yang nantinya digunakan seperti, wawancara, survei, dan *workshop*.
- Menentukan teknik analisis kebutuhan seperti, *use case*, *flowchart*, dan diagram alir.
- Menentukan dan mempelajari mengenai desain sistem seperti diagram UML, ERD, dll.

3. Pengerjaan Proyek (Analisis) 4 Minggu

- Membuat perencanaan proyek, seperti menentukan tujuan, ruang lingkup atau batasan-batasan dalam pengembangan proyek.
- Pengumpulan kebutuhan dengan melakukan wawancara dan survei dengan pemangku kepentingan.

- Analisis kebutuhan dengan mendokumentasikan dan menganalisis kebutuhan yang telah dikumpulkan.
4. Pengerjaan Proyek (Desain, Pengembangan, dan Testing) **12 Minggu**
- Membuat tampilan desain sistem berdasarkan kebutuhan yang telah dianalisis.
 - Membuat Rest API *Backend*.
 - Membuat dokumentasi lengkap dan desain sistem kemudian menguji dengan pengguna untuk mendapatkan umpan balik.
5. Penyelesaian dan Penilaian **1 Minggu**
- Pengujian akhir sistem secara menyeluruh untuk memastikan semua kebutuhan telah terpenuhi.
 - Presentasi hasil proyek kepada tim dan instansi magang.
 - Penilaian oleh mentor tentang kinerja, keterampilan yang diperoleh, dan kontribusi selama magang berlangsung.
6. Penutupan **1 Hari**
- Ringkasan pengalaman magang dan pelajaran yang telah didapat.
 - Penyerahan aplikasi kepada instansi magang
 - Pemberian sertifikat magang.