

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Organisasi

Software Programmer di dalam PT HashMicro Solusi Indonesia mendapat tugas membuat perangkat lunak sebagai *Backend Engineer* sekaligus *Software Engineer*. *Software Programmer* akan sangat sering bekerja bersama *System Analyst* dan *Software Programmer* lainnya, serta tidak jarang juga akan berhadapan langsung dengan *Client*. Di dalam PT HashMicro Solusi Indonesia *Software Programmer* dibagi lagi menjadi dua tim, yaitu tim produk, dan tim proyek. Tim produk bertugas membangun *pre-built/core ERP software* sebelum dikembangkan lebih lanjut oleh tim proyek. Tim proyek akan melakukan pengembangan *software* sesuai permintaan perusahaan/*Client* (kostumisasi).

Sebagai tim proyek *Software Programmer* dalam PT HashMicro Solusi Indonesia, pengembangan *software* dilakukan diatas program *core* yang sebelumnya telah dibuat oleh tim produk. Pengembangan *software* yang berdasar dari *core* tersebut nantinya akan disesuaikan dengan metode modifikasi dan/atau dilakukan penambahan fitur sesuai kebutuhan dan permintaan *Client*.

3.2 Tugas yang Dilakukan

Sebagai *Software Programmer* dalam divisi proyek PT Hashmicro Solusi Indonesia ada dua jenis tugas yang dikerjakan, antara lain:

1. Tugas umum *Software Programmer*
 - (a) Menulis code dengan kerangka kerja Odoo.
 - (b) Bekerja sama dengan tim pengembangan untuk menentukan kebutuhan aplikasi.
 - (c) Mendesain, mengoptimalkan, dan menganalisis struktur *database*.
 - (d) Mengembangkan komponen *backend*.
 - (e) Meningkatkan fungsionalitas *database*.
 - (f) Men-*debug* dan melakukan *application testing*
2. Tugas *Software Programmer* divisi proyek

- (a) Melakukan koordinasi dengan tim analis.
- (b) Melanjutkan pengembangan dari tim produk.
- (c) Melakukan modifikasi modul.
- (d) Menambah, memperbaiki, dan meningkatkan fitur.
- (e) Memenuhi kebutuhan klien.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang diuraikan seperti pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	<i>Onboarding</i> dan Uji Kompetensi berdasarkan <i>role</i>
2	Pembelajaran Git, Python, dan instalasi Odoo
3	Pembekalan konsep, paradigma, contoh kasus, <i>design pattern</i> , dan fitur - fitur yang akan dikerjakan dengan <i>Framework</i> Odoo
4	Pembelajaran <i>Framework</i> Odoo
5	Pengerjaan project (modul) individu dan pembelajaran Odoo
6	Persiapan kebutuhan bekerja dan melakukan perubahan terhadap modul Accounting dan modul Payment
7	Penambahan fitur <i>state</i> dan reject action dalam modul <i>Cash Advance</i>
8	Penambahan fitur Department Approval pada modul <i>Cash Advance</i> dan implementasi fitur notifikasi pada modul <i>Expenses</i>
9	Melakukan perubahan <i>fields</i> dan <i>bug fixing</i> pada modul <i>Expenses</i>
10	Melakukan perubahan terhadap fitur notifikasi dan <i>bug fixing</i> pada modul <i>Expenses</i>
11	Menambah fitur <i>filter</i> dan <i>domain</i> pada modul <i>Expenses</i>
12	Melakukan penyesuaian terhadap fitur notifikasi dalam modul <i>Expenses</i>
13	Melakukan perbaikan terhadap modul Purchase Request
14	Melakukan perubahan <i>state</i> dan menu pada modul <i>Cash Advance</i> dan penambahan fitur <i>summary</i> pada modul <i>Expenses</i>

15	Melakukan perubahan dan penyesuaian terhadap modul Accounting
16	Melakukan perubahan dan penyesuaian terhadap keseluruhan proyek dan penambahan <i>pre-processing</i> terhadap respon Purchase Status
17	Melakukan penyesuaian terhadap modul <i>Expenses</i> dan penambahan <i>post-processing</i> pada <i>payment messages</i>
18	Penambahan grup konfigurasi
19	Melanjutkan pengembangan dan <i>maintenance</i> proyek
20	Melakukan penyesuaian modul <i>Cash Advance</i> dengan modul <i>Expenses</i>

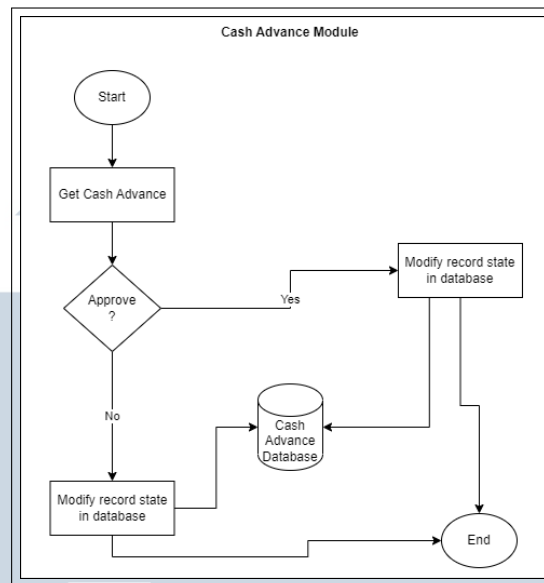
3.3.1 Keterangan Pekerjaan

Software Programmer PT Hashmicro Solusi Indonesia, terutama dalam divisi proyek berkewajiban untuk menyesuaikan aplikasi yang dikembangkan oleh divisi produk untuk sesuai dengan kebutuhan perusahaan klien. Pekerjaan utama yang diberikan pada penulis antara lain:

A. Penyesuaian modul *Cash Advance*

Modul *Cash Advance* adalah modul untuk membuat dan menambahkan jenis pengeluaran yang akan diajukan seorang pegawai atau suatu departemen dalam sebuah perusahaan. Modul ini disesuaikan dengan kebutuhan perusahaan dari klien, penyesuaian tersebut antara lain:

1. Penambahan fitur *Cash Advance Approval*. Fitur ini berfungsi untuk seseorang dapat melakukan persetujuan apakah *Cash Advance* yang ditambahkan sudah sesuai dan memenuhi kriteria atau tidak.



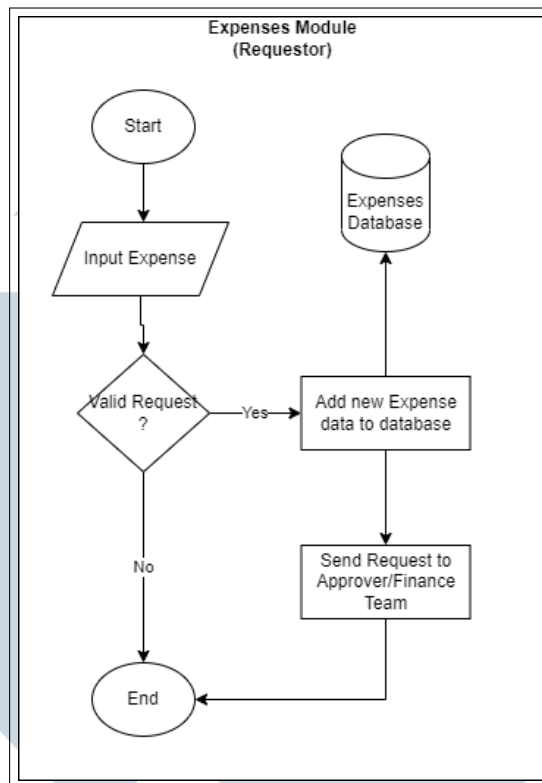
Gambar 3.1. Flowchart dari alur fitur *Cash Advance Approval*

2. Penambahan fitur *Cash Advance Summary*. Fitur *Cash Advance Summary* merupakan fitur tambahan berupa *field* yang berisi judul atau jenis dari *Cash Advance* yang baru ditambahkan, sehingga mempermudah dalam mengidentifikasi dan mengklasifikasikan suatu *Cash Advance*.

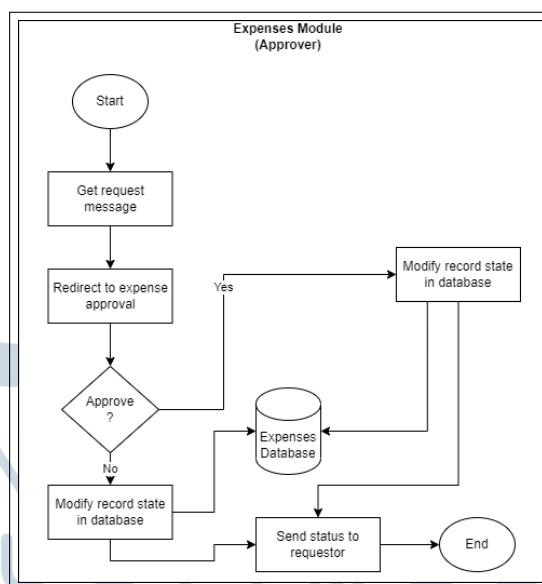
B. Pengembangan modul *Expenses*

Modul *Expenses* adalah modul untuk menjalankan mekanisme sekaligus memantau pengeluaran perusahaan baik dilakukan oleh individu maupun antar departemen. Fitur-fitur yang telah ditambahkan dan disesuaikan dengan perusahaan klien adalah sebagai berikut.

1. Melakukan peningkatan dan penyesuaian modul *Expenses* dengan menambah fitur *Domain Filter (Expense Category)*. Fitur *domain filter* berfungsi untuk membagi jenis *expense* ke dalam berbagai kategori, sehingga *Expenses* dapat ditampilkan pada user sesuai *expense's state* itu sendiri.
2. Menambah fitur *Request as Notification*. Fitur *Expense Request as Notification* adalah fitur untuk mengirim *expense request* ke setiap inbox dari pegawai yang memberi persetujuan dan/atau tim keuangan sehingga tim pemberi persetujuan dapat langsung dialihkan ke fitur persetujuan.



Gambar 3.2. Flowchart dari alur fitur *Request as Notification requestor side*

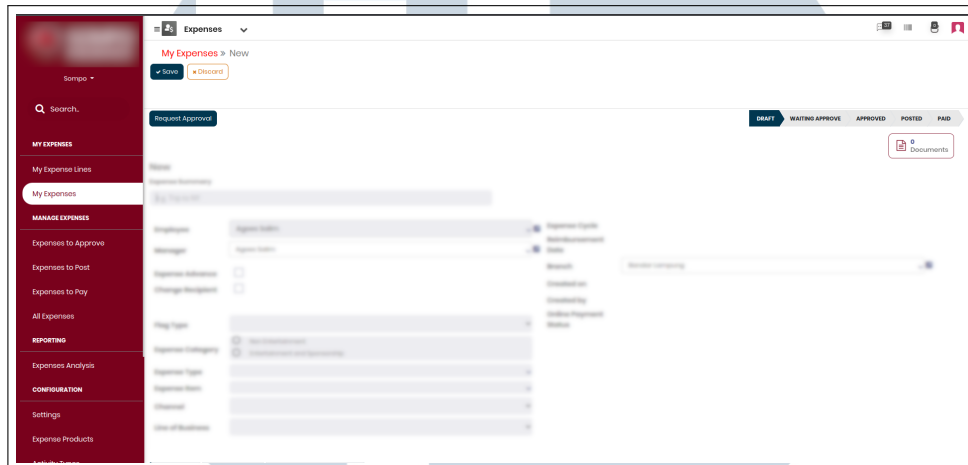


Gambar 3.3. Flowchart dari alur fitur *Request as Notification approver side*

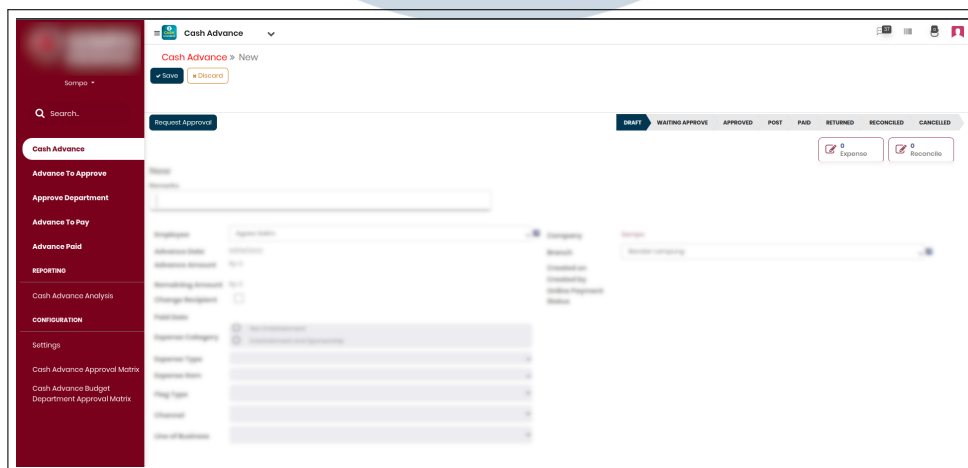
3.3.2 Hasil Pengembangan

Pengembangan dan penyesuaian fitur dalam modul *Expenses* sudah mulai memenuhi kebutuhan dari klien, tetapi masih harus dilakukan berbagai uji coba

sehingga fitur-fitur tersebut nantinya masih harus dimasukkan ke dalam tahap pemeliharaan dan terus dikembangkan dan disesuaikan. Banyak fitur yang masih ingin diimplementasikan dan diujicoba oleh klien sehingga proyek juga akan terus berkembang seiring waktu. Berikut merupakan tampilan dari hasil modul *Expenses* dan *Cash Advance*.



Gambar 3.4. *Expenses*

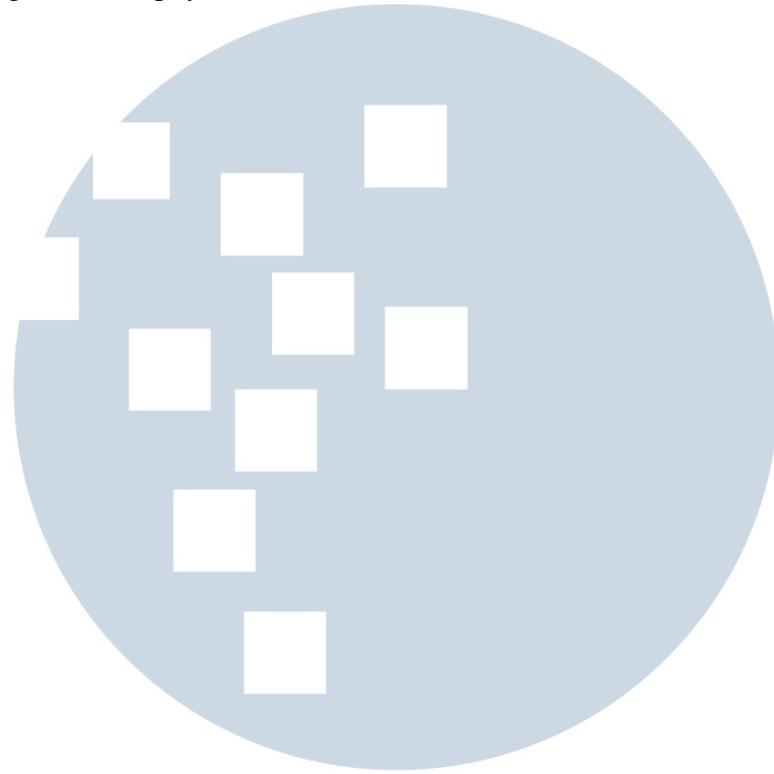


Gambar 3.5. *Cash Advance*

3.4 Kendala dan Solusi yang Ditemukan

Pengerjaan fitur dilakukan secara *trial & error* sehingga penamaan dan struktur kode tidak beraturan. Hal yang harus dilakukan adalah memasukan fitur ke tahap pemeliharaan dan melakukan *Code Refactoring*. *Code Refactoring* adalah suatu kegiatan peningkatan dan penyederhanaan code tanpa membuat

fungsionalitas baru [5]. Sehingga kode lebih rapih dan lebih mudah untuk dilakukan pengembangan terhadapnya.



UMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA