

Analisa dan Perancangan Warehouse Inventory System Untuk UMKM Berbasis Multi Tenant

Wiyoga Baswardono

Jurnal Algoritma
Sekolah Tinggi Teknologi Garut
Jl. Mayor Syamsu No. 1 Jayaraga Garut 44151 Indonesia
Email: jurnal@sttgarut.ac.id

wiyoga.b@sttgarut.ac.id

Abstrak – Manajemen yang baik dalam bisnis, sangat di perlukan karena akan mempengaruhi kinerja yang baik dalam bisnis tersebut. Salah satu tipe bisnis di indonesia adalah UMK. Pada tipe bisnis ini biasanya ada barang yang harus di kelola, baik untuk keperluan pribadi atau umum untuk kepentingan bisnis itu sendiri. Dalam pengaturan untuk barang-barang tersebut di butuhkan gudang atau tempat untuk menyimpan barang atau inventaris dari bisnis tersebut. Tujuan dari penelitian ini adalah menganalisis dan desain sistem inventaris gudang (warehouse inventory system) untuk membantu pengelolaan barang atau inventaris. Dalam Sistem ini dapat dapat mengetahui barang yang tersedia, pengelolaan stok barang dan fungsi lainnya yang membantu pengelolaan setiap gudang pada bisnis tersebut. Untuk desain multitenant pada sistem yang di rancang ini di sebabkan karena bisa salah satu pemilik bisnis dapat memiliki bisnis yang berbeda - beda , tidak hanya satu macam, di karena itu di butuhkan fungsi bisnis dan proses yang berbeda serta kustomisasi untuk memecahkan masalah tersebut.

Kata Kunci – analysis, design, inventory, multitenant, warehouse

I. PENDAHULUAN

Perkembangan bisnis sekarang membuat persaingan semakin ketat dalam dunia usaha, ketepatan dan kecepatan merupakan hal yang utama dalam bertindak. Pengelolaan yang baik pada suatu bisnis diperlukan untuk memperlancar kinerja dalam bisnis tersebut. Salah satu pelaku bisnis di Indonesia adalah Usaha Mikro, Kecil dan Menengah (UMKM).

Usaha Mikro, Kecil dan Menengah (UMKM) memiliki peranan penting dalam perekonomian di Indonesia. UMKM memiliki proporsi sebesar sebanyak 56-54 juta unit dari total keseluruhan pelaku usaha di Indonesia [1]. Usaha Mikro, Kecil, dan Menengah telah mampu membuktikan eksistensinya dalam perekonomian di Indonesia. Dalam bisnis UMKM (usaha mikro kecil menengah) terdapat berbagai macam jenis macam usaha seperti pada bidang kuliner, pendidikan otomatis dan lain-lain. Pada setiap usaha tersebut biasanya memiliki barang yang harus di kelola, baik itu untuk kebutuhan bisnis atau pribadi dari setiap usaha tersebut. Dalam pengelolaan suatu barang tersebut untuk usaha tersebut di butuhkan warehouse (gudang) atau tempat untuk menyimpan barang atau *inventory*. Dikarenakan itu, dibutuhkan suatu sistem untuk membantu pengelolaan barang dalam *warehouse* tersebut.

Sistem *Warehouse Inventory System* ini di digunakan untuk membantu pengelolaan barang atau *inventory*. Pada sistem ini dapat mengetahui barang yang di digunakan pada usaha tersebut, serta dapat mengetahui stok dari barang tersebut pada setiap *warehouse*, serta dapat melakukan *adjustment*, *manage* barang, pembelian (*purchasing*) barang ke *supplier*, transfer barang dari satu *warehouse* ke *warehouse* lainnya [2]. Dikarenakan sistem ini di gunakan untuk usaha, maka di sini di ajukan di buat dengan *multitenant*.

Perancangan *Multitenant* di buat karena dapat satu pengusaha memiliki beberapa usaha yang

berbeda dan tentu membutuhkan kebutuhan warehouse yang berbeda sehingga di perlukan nya kustomisasi. Dalam satu usaha dengan usaha yang lain dapat memiliki *database* yang berbeda, dengan satu aplikasi yang sama tetapi *owner* (pengusaha atau perusahaan) sama. Maka dengan permasalahan yang ada, manfaat yang diharapkan dari penelitian ini adalah sebagai dasar perancangan dan acuan tahapan pembuatan dan implementasi dari sistem ini.

II. TINJAUAN TEORITIS

A. Sistem Informasi

Sistem informasi merupakan kombinasi yang terorganisir antara pengguna, perangkat keras, perangkat lunak, jaringan komunikasi, sumber daya data kebijakan prosedur yang menyimpan, mengambil, mengubah, menyebarkan informasi dalam sebuah organisasi [3, 4, 5, 6].

B. Warehouse

Warehouse (gudang) merupakan fasilitas khusus yang bersifat tetap, yang dirancang untuk mencapai target tingkat pelayanan dengan total biaya yang paling rendah. Manajemen pergudangan dirancang bertujuan untuk mengontrol kegiatan pergudangan yang diharapkan dari pengontrolan ini adalah terjadinya pengurangan biaya-biaya yang ada di dalam gudang, pengambilan dan pemasukan barang ke gudang yang efektif dan efisien, serta kemudahan dan keakuratan informasi stok barang di gudang [7].

C. Inventory

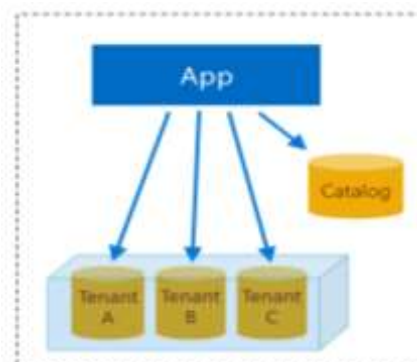
Persediaan atau inventory adalah barang yang di peroleh dan tersedia dengan tujuan untuk di lakukan proses dijual atau di pakai dalam produksi atau di pakai untuk keperluan non produksi dalam siklus kegiatan yang normal [8].

D. Usaha Mikro, Kecil dan Menengah (UMKM)

Karakteristik UMKM merupakan sifat atau kondisi faktual yang melekat pada aktifitas usaha maupun perilaku pengusaha yang bersangkutan dalam menjalankan bisnisnya. Karakteristik ini yang menjadi ciri pembeda antar pelaku usaha sesuai dengan skala usahanya. Menurut Bank Dunia, UMKM dapat dikelompokkan dalam tiga jenis, yaitu: 1. Usaha Mikro (jumlah karyawan 10 orang); 2. Usaha Kecil (jumlah karyawan 30 orang); dan 3. Usaha Menengah (jumlah karyawan hingga 300 orang) [9].

E. Multi Tenant

Multitenant merupakan suatu prinsip dari arsitektur perangkat lunak, dimana sebuah perangkat lunak yang berjalan di atas server melayani banyak pengguna / tenant. Dengan prinsip multitenant ini, sebuah perangkat lunak dirancang untuk memiliki partisi data yang berbeda dan dapat dikonfigurasi [10].



Gambar 1: *Multi-tenant app with database-per-tenant*

Penggunaan aplikasi multi-tenant dengan banyak *database*, database di sediakan untuk setiap tenant yang baru. Aplikasinya dapat di skalakan secara horizontal dengan bertambahnya lebih banyak lagi *node*. *Scalability* di dasarkan pada beban kerja dan tidak tergantung pada jumlah atau besar dari masing-masing *database*.

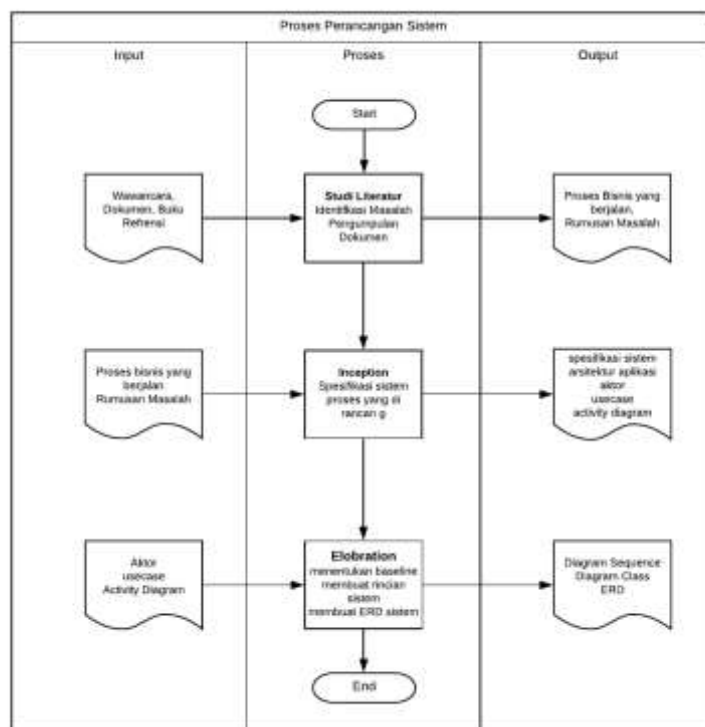
Sama seperti halnya aplikasi biasanya, penggunaan *single-tenant databases*, penggunaannya memberikan isolasi untuk *tenant* tersebut. Schema yang di berikan dari tiap database dapat di kustomisasi dan di optimasi untuk tenant tertentu. Kustomisasi ini tidak memberikan efek terhadap tenant lain dalam aplikasi dan tentunya ada kemungkinan data field di perlukan *indexing*. Dengan *database – per-tenant*, kustomisasi schema untuk satu atau lebih *tenants* lebih memungkinkan untuk di capai.

F. Rational Unified Process (RUP)

Rational Unified Process (RUP) RUP adalah sebuah proses perancangan perangkat lunak yang menggunakan pendekatan iterative, architecture-centric, dan use-case-driven dirumuskan dengan baik dan disusun dengan baik [11]. Ini tergambar dengan jelas siapa yang bertanggung jawab untuk tugasnya, bagaimana berbagai hal selesai, dan kapan untuk melakukannya. RUP sendiri telah di desain dengan teknik yang mirip dengan yang digunakan pada desain perangkat lunak. Untuk proses pemodelan yang baku menggunakan Unified Model Language (UML), Business Model process Modelling (BPMN) dan Entity Relation Diagram (ERD).

III. METODE PENELITIAN

Pada proses perancangan sistem yang akan dilakukan, berikut adalah tahapan aktivitas yang akan dilakukan dan dapat digambarkan pada kerangka kerja konseptual, dengan mengacu pada metode pengumpulan data dan metode pengembangan sistem yang digunakan.



Gambar 2: Proses Perancangan Sistem

A. Studi Literatur

Tahapan ini dilakukan untuk mengidentifikasi komponen-komponen yang terkait pada penelitian dalam *Warehouse Inventory System*, seperti dengan melakukan nya observasi, studi

dokumen dan wawancara dengan beberapa pihak yang membutuhkan sistem ini serta di sesuaikan dengan kebutuhan untuk Usaha Mikro, Kecil dan Menengah (UMKM) secara umum, maupun identifikasi kebutuhan untuk rancangan untuk di buat konsep *multi tenant*. Untuk mengetahui proses bisnis yang berjalan dan rumusan masalah.

B. Inception

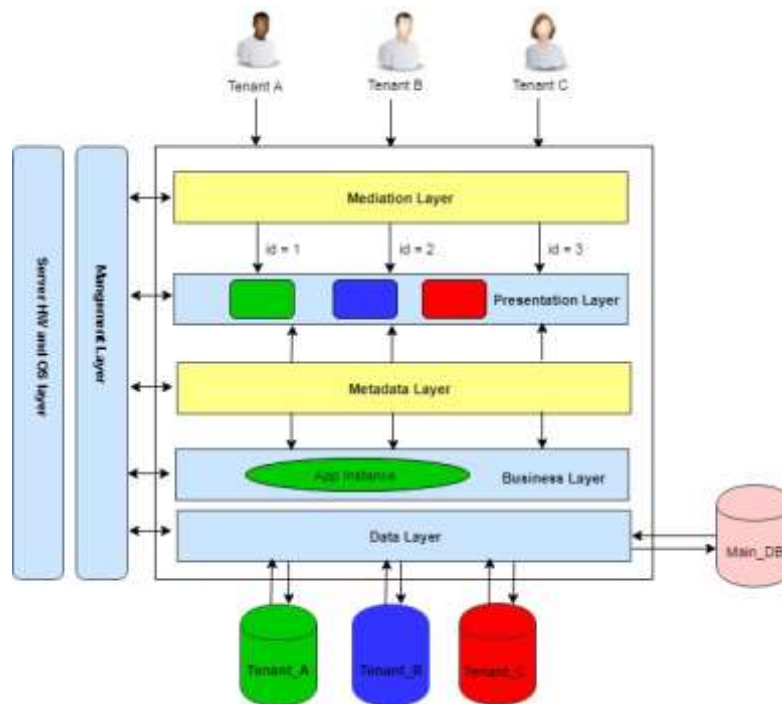
Tahapan inception merupakan tahap awal, tahap pengenalan atau tahap persiapan. Dalam tahapan ini dilakukan beberapa aktifitas seperti mengidentifikasi kebutuhan skema dan sistem, mengidentifikasi aktor, pembangunan atau pembuatan proses bisnis, dan planning (perencanaan) dari proyek, dan ada tahap ini akan menghasilkan use case diagram, activity diagram untuk menggambarkan proses *Warehouse Inventory System* dengan konsep multitenant tersebut [12, 13].

C. Elaboration

Pada tahap elaboration ini akan dilakukan analisis lebih lanjut dari tahapan inception. Adapun hasil yang didapat dari inception, dan akan dilanjutkan pada tahapan elaboration seperti, membuat baseline yang akan menghasilkan class diagram, entity relation diagram untuk menggambarkan basis data yang terjadi dalam *Warehouse Inventory System* [14].

HASIL DAN DISKUSI

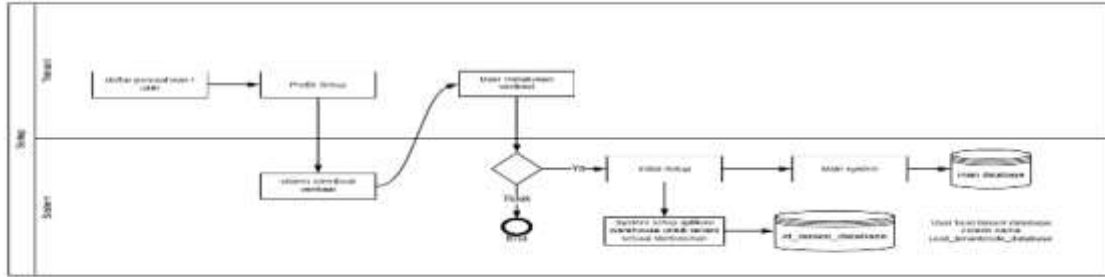
A. Perancangan Arsitektur



Gambar 3: Arsitektur Aplikasi

Pada arsitektur aplikasi tersebut, Tenant terhubung dengan layer yang memfasilitasi komunikasi antara servis yang berbeda (*mediation layer*) melalui *Id unique* yang terdapat pada *tenant* tersebut terhubung dengan layer mengolah *delivery* dan *formatting* untuk diproses atau ditampilkan (*presentation layer*), yang diolah atau representasikan secara sendiri untuk setiap *tenant*. Terhubung dengan *metadata layer* (yang berfungsi sebagai yang menyediakan informasi data yang masuk baik itu dari *presentation layer* atau *data layer*), serta terhubung ke *app instance* dan *business layer*, serta kedua item tersebut terhubung *data layer* yang terhubung ke database. *Data database* di terbagi dua jenis *main_db* untuk pengaturan data tenant yang akan menggunakan sistem serta nanti untuk penambahan atau pengurangan aplikasi fitur di atur di sini dan *tenant_db* ini untuk mengatur data yang akan di gunakan transaksi tenant tersebut dalam hal di sini adalah data *warehouse* atau *inventory*.

B. Business Process Wis (Warehouse Information System)

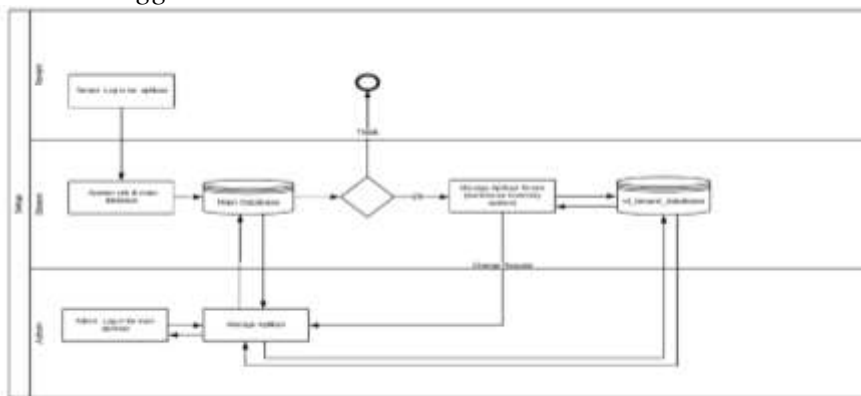


Gambar 4: Flowchart Business Process Warehouse Information system

Keterangan mengenai **Business Proses WIS** ini:

1. Applicant daftar sebagai Tenant (untuk pribadi) atau perusahaan, setelah itu melakukan *profile setup* di sini pencatatan kebutuhan dan data ini untuk kebutuhan analisa untuk pengembangan sistem ini.
2. Setelah pendaftaran sistem melakukan verifikasi, verifikasi tersebut dapat di tolak secara langsung, yang berhasil di kirimkan kembali verifikasi berupa email atau cara lain. Apabila user melakukan verifikasi kembali maka akan di lanjutkan ke tahap berikutnya , apabila tidak maka berakhir biasanya tenant di berikan batas waktu 2 hari untuk melakukan verifikasi kembali apabila tidak, data di simpan dan tidak ditindaklanjuti.
3. Apabila Tenant melakukan verifikasi maka sistem akan melakukan initial setup, pada main system data tenant tersebut akan di input ke dalam main database serta pada main system data tenant tersebut sudah dapat di olah. Pada Initial setup sistem melakukan untuk sistem WIS sesuai dengan kebutuhan tenant tersebut, serta sistem membuat database baru khusus untuk pengolahan data tenant tersebut. Contoh nanti pada nama database `UUID_tenantcode_database`. *UUID (Universally unique identifier)* ini di gunakan supaya lebih unik karena nanti kebutuhan diperkirakan semakin besar, tenantcode ini di sesuaikan dengan pengkodean yang akan di lakukan oleh system. Setelah sistem di setup maka Tenant/user sudah dapat sign in atau masuk ke dalam sistem.

C. Business Process Penggunaan WIS



Gambar 5: Flowchart Business Process Penggunaan Information system

Keterangan mengenai **Business Proses penggunaan WIS** ini:

1. Tenant sign in / login ke aplikasi system cek sesuai kondisi yang nanti di ditetapkan oleh admin.misalkan apakah tenant masih free user dalam kurung waktu 30 hari apabila masih free user dapat masuk apabila tidak bisa di login. *Rule* semua ini disetting pada main system. Maka nanti harus disiapkan parameter untuk pengaturan ini. Apabila user atau tenant tidak dapat login maka tidak dapat menggunakan sistem WIM tersebut, apabila user atau tenant bisa login atau sign in maka dapat sistem WIM tersebut dan juga dapat melakukan change request apabila di butuhkan yang tertuju pada main aplikasi.

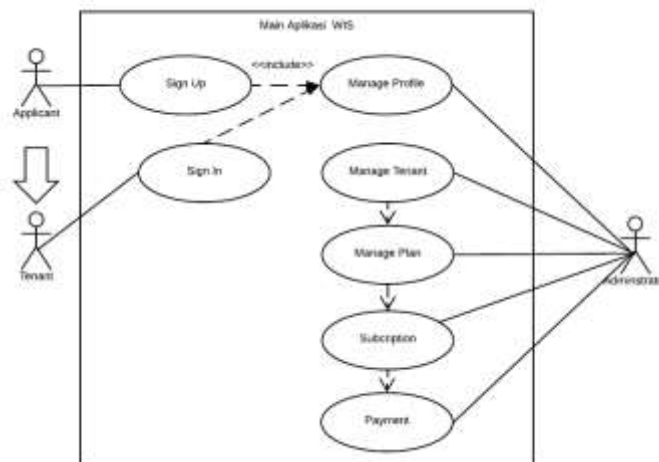
2. Admin dari sistem ini hanya dapat melakukan perubahan lewat main aplikasi untuk setiap sistem WIM yang di buat pertainant tersebut. Tetapi dapat melakukan suatu perubahan ke sistem atau perubahan ke database apabila ada change request / permintaan perubahan dari tenant/user. Setiap tenant bisa mempunyai perubahan kebutuhan sesuai yang di butuhnya.

D. Use Case

Use case menggambarkan serangkaian kegiatan yang di lakukan untuk menghasilkan hasil (*output*) dari kasus yang di gambarkan. Setiap penggunaan menggambarkan bagaimana *user* eksternal memicu suatu event yang membuat sistem merespon [15, 16, 17, 18].

Pada Perancangan usecase di sini terdapat 2 aplikasi: Main aplikasi dan tenant aplikasi, main aplikasi itu untuk mengatur tenant sedangkan tenant aplikasi untuk sistem WIS yang di pergunakan oleh tenant sendiri. Untuk mempermudah main aplikasi kodenya adalah MA dan tenant aplikasi adalah TA.

Berikut adalah use case dari main aplikasi *warehouse information system*:



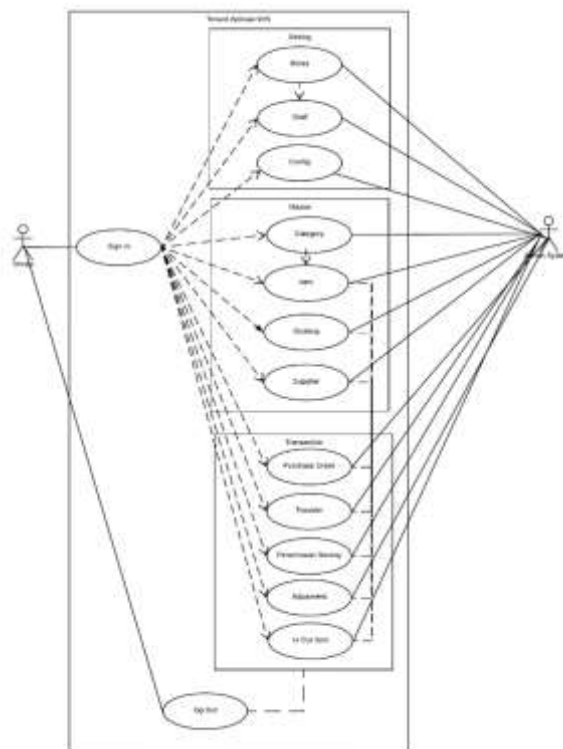
Gambar 6: Use case Main aplikasi *warehouse information system*

Tabel 1 Use Case Sign up / Sign in

Use Case ID:	MA-001		
Use Case Name:	Sign up / Sign in		
Created By:	-	Last Updated By:	-
Date Created:	-	Date Last Updated:	-
Actor:	Applicant , Tenant		
Description:	Pada usecase ini saya jadikan sign up -> sign in. sebelum sign up (mendaftar) maka aktor yang daftar adalah applicant .apabila telah mendaftar maka yang sign adalah Tenant. Untuk input ke dalam sistem. tetapi pada Main sistem ini tenant hanya dapat masuk untuk mengubah profil. Untuk sekarang dalam rancangan permintaan perubahan sistem di lakukan secara manual (lewat email, tlp dan lain-lain)		
Preconditions:	Sign up untuk sign in		
Postconditions:	Manage profile		
Priority:	-		
Frequency of Use:	Sign up (sekali saja), sign in (sering)		
Normal Course of Events:	Pada sign up terdapat verifikasi dari sistem apabila verifikasi tidak di lanjutkan oleh applicant maka sign up gagal. Pada sign in terdapat verifikasi login sesuai dengan rule yang di		

	buat oleh sistem. bahwa tenant tersebut dapat masuk atau menggunakan sistem pada WIS
Alternative Courses:	-
Exceptions:	-
Includes:	-
Special Requirements:	Dari sistem mendeteksi user dari tenant tersebut mana yang sign in ke main aplikasi , dan mana yang sign in ke tenant aplikasi untuk penggunaan aplikasi
Assumptions:	Dalam asumsi dalam perancangan ini satu sign in tenant masih merupakan satu aplikasi yang di gunakan , belum beberapa
Notes and Issues:	Rule bisnis atau peraturan penggunaan sistem sebaiknya telah di tentukan terlebih dahulu

Berikut adalah use case dari **Tenant aplikasi warehouse information system** :



Gambar 7: Use case Tenant aplikasi warehouse information system

Tabel 2 Use Case Manage Staff

Use Case ID:	TA-002		
Use Case Name:	Staff		
Created By:	-	Last Updated By:	-
Date Created:	-	Date Last Updated:	-
Actor:	Tenant, Admin System		
Description:	Manage staff, untuk mengatur staff / user yang akan menggunakan aplikasi WIS .dari form ini bisa input staff baru beserta parameter yang telah di tentukan, tentu juga membuat user tersebut untuk sign in untuk menggunakan aplikasi tersebut. Terlepas itu staff ini terhubung dengan roles. Untuk menentukan parameter bahwa staff tertentu dengan user bisa memakai modul yang di butuhkan		

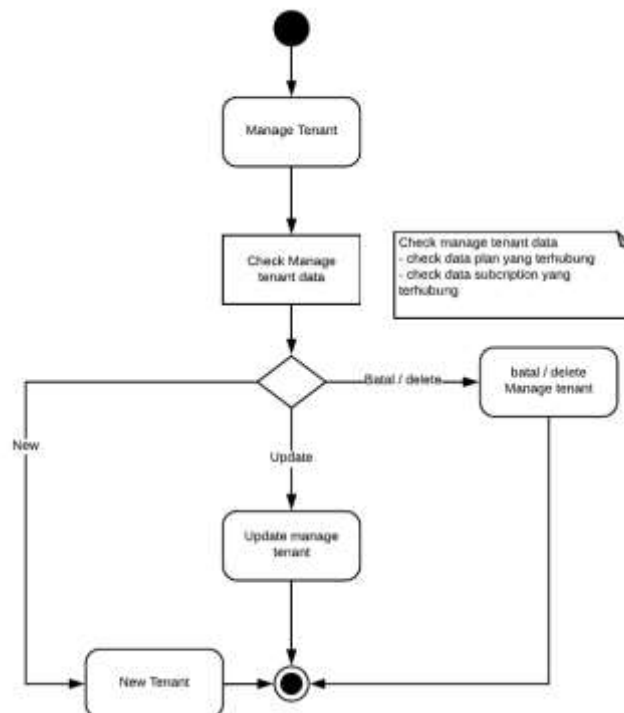
	saja.tentu yang sudah di tentukan pada roles.
Preconditions:	Roles
Postconditions:	-
Priority:	Roles harus di buat terlebih dahulu , sebaiknya ini salah satu modul yang di buat pertama sebelum yang lain
Frequency of Use:	Sering
Normal Course of Events:	Admin input/ update untuk membuat/update data staff. pada data staff tersebut di tentukan nama user beserta password untuk kebutuhan sign in kepada sistem ,dan di tentukan rolesnya di tentukan sesuai dengan deskjob masing-masing (kebutuhannya)
Alternative Courses:	-
Exceptions:	-
Includes:	Roles
Special Requirements:	Password di encrypt. Username tidak bisa sama, roles untuk superadmin di sembunyikan (hardcode)
Assumptions:	-
Notes and Issues:	Penentuan dan desain <i>User Interface</i> di sini nanti sangat menentukan efesiensi dari rancangan ini.

E. Activity Diagram

Activity Diagram memodelkan perilaku yang ada pada bisnis proses. Menunjukkan *flow control* dan *flow data* [19].

Pada Perancangan Activity Diagram di sini terdapat 2 aplikasi: Main aplikasi dan tenant aplikasi, main aplikasi itu untuk mengatur tenant sedangkan tenant aplikasi untuk sistem WIS yang di pergunakan oleh tenant sendiri. Untuk mempermudah main aplikasi kodenya adalah MA dan tenant aplikasi adalah TA.

F. Main Aplikasi Activity Diagram



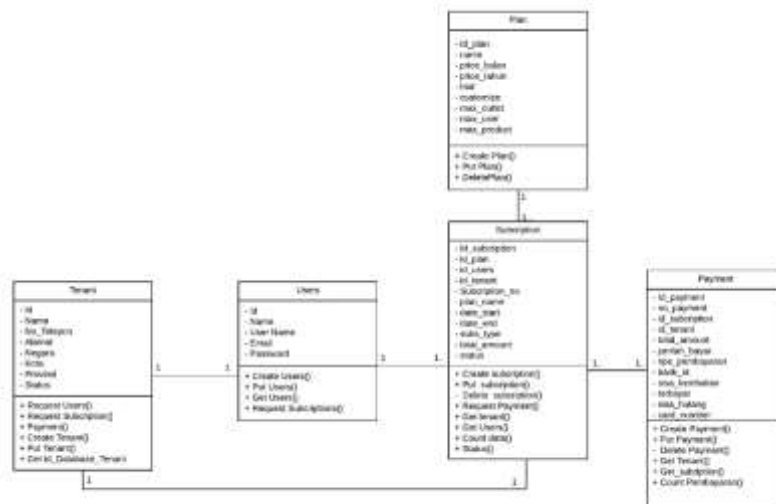
Gambar 8: Activity Diagram Tenant

Tabel 3 Use Case Manage Tenant

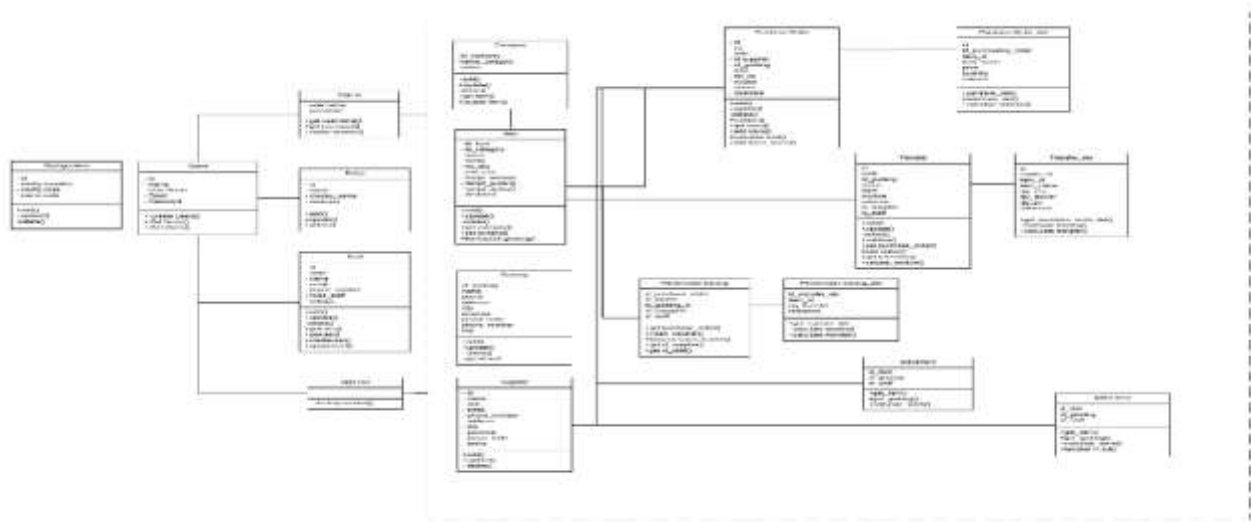
No : MA-003	
Nama :	Manage Tenant
Tujuan	Untuk manage tenant beserta plan, setting, subscription serta history pembayaran untuk tenant tersebut
Deskripsi	
Tipe	
Aktor	Admin
Skenario Utama	
Kondisi Awal	
Aksi Aktor	Reaksi Sistem
1. User membuka dan cek form tenant	
	2. Sistem cek list tenant yang ada dalam sistem beserta parameter nya yang lain
3. User memilih tenant yang akan di lakukan aksi ya. Aksi terbagi 3 : <ul style="list-style-type: none"> • Menambah tenant baru • Edit tenant • Hapus tenant 	
	4. Aksi yang di lakukan sistem apabila user memilih : <ul style="list-style-type: none"> • Menambah tenant, ini membuat tenant baru tanpa ada signup. Ini di buat untuk keperluan khusus apabila di butuhkan. • Edit tenant, sistem akan update data yang di input oleh user untuk update data tersebut • Hapus tenant, sistem akan menghapus data yang di pilih oleh user (ini akan mempengaruhi parameter yang lain maupun siste)
Kondisi Akhir	Adanya perubahan pada data manage tenant

G. Class Diagram

Class Diagram adalah model statis untuk mendukung pandangan secara statis dari sistem yang berkembang. Pada class diagram menunjukkan hubungan di antara class yang tetap konstan di dalam sistem yang di rancang.



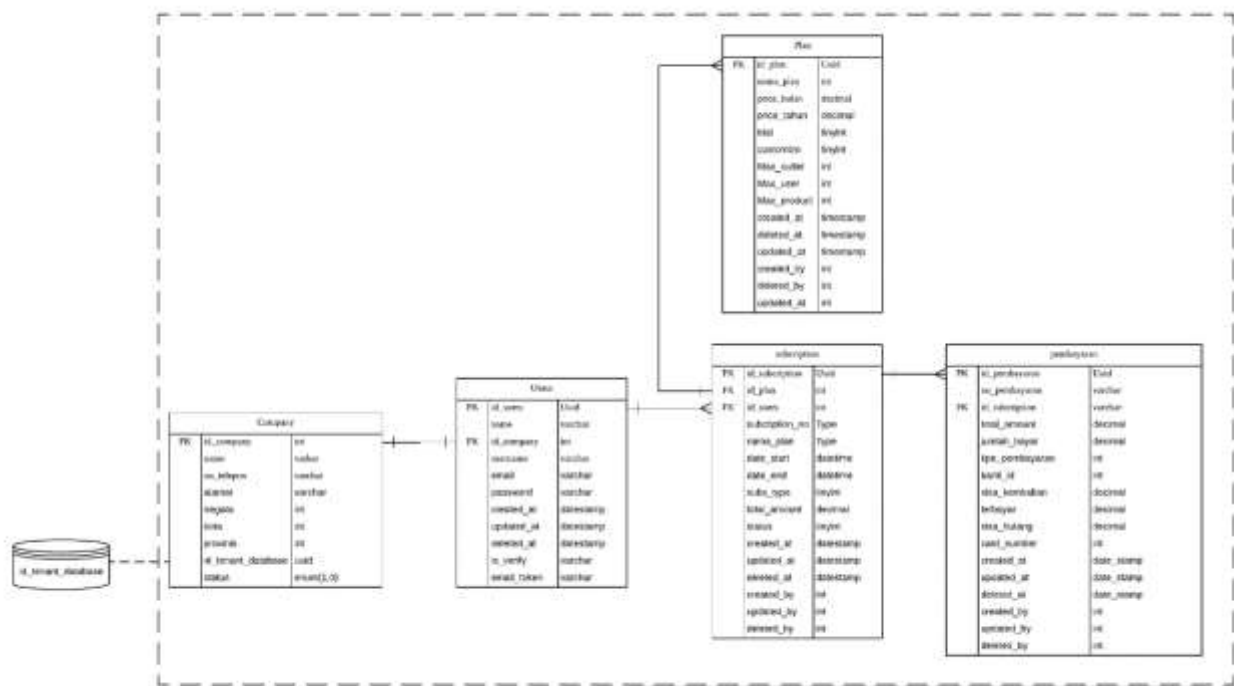
Gambar 9: Class Diagram Master Aplikasi Warehouse Information System



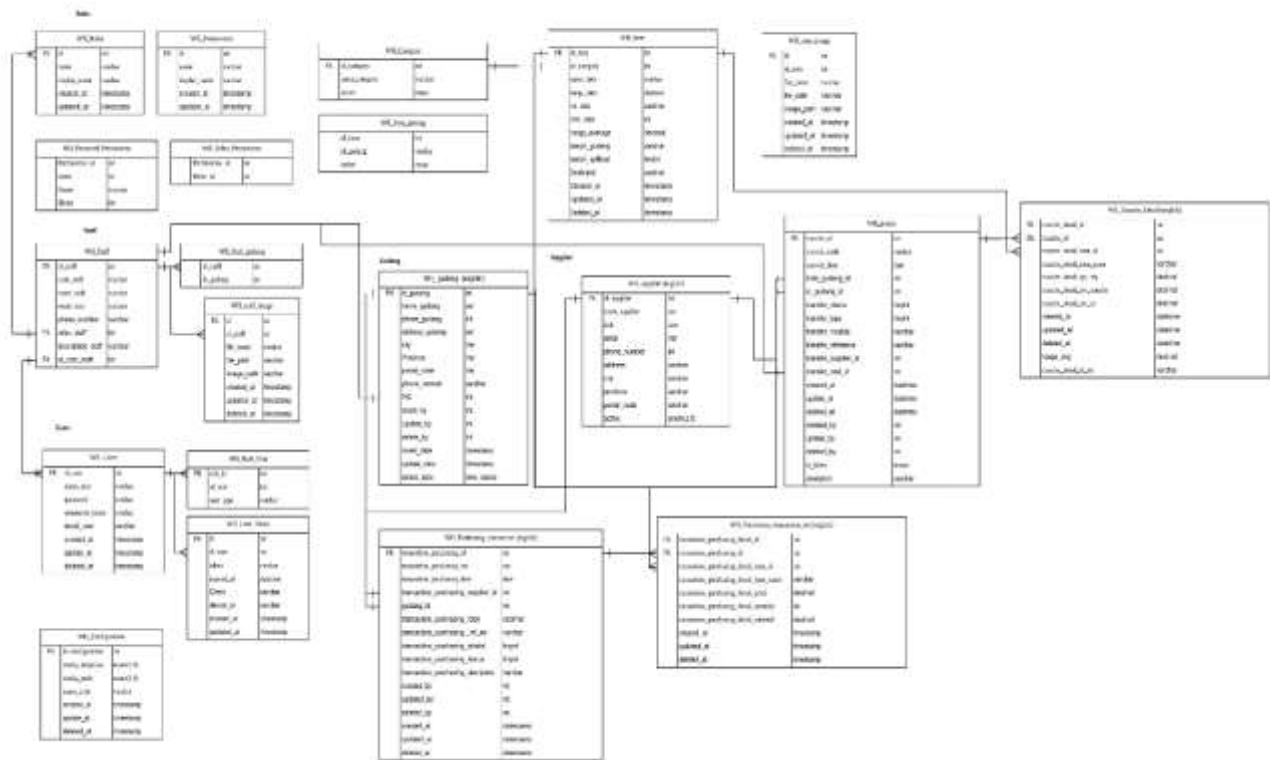
Gambar 10: Class Diagram Tenant Aplikasi Warehouse Information System

H. ERD (Entity Relation Diagram)

Entity Relation Diagram adalah gambar yang dibuat untuk menunjukkan informasi, disimpan dan digunakan oleh bisnis sistem. Dari Entity Relationship Diagram (ERD) di harapkan dapat di temukan informasi yang akan terdapat pada sistem dan bagaimana mereka terorganisir dan berelasi satu dengan yang lain [20].



Gambar 11: Entity Relation Diagram Main Aplikasi Warehouse Information System



Gambar 12: Entity Relation Diagram Tenant Aplikasi Warehouse Information System

KESIMPULAN

Berdasarkan hasil analisis dan perancangan *Warehouse Inventory System* ini, maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. *Business rule* (aturan bisnis) pada analisa dan perancangan WIS ini sangat berpengaruh terhadap nanti pembuatan aplikasi ini, kebutuhan secara *Hardware* nanti ke depan nya sebaiknya pada teknologi *cloud*..Karena kebutuhan ini sangatlah *flexible* tidak menutup ke depan nya kebutuhan akan semakin besar atau tidak sama sekali. Serta dalam aturan bisnis pasti akan terjadi perubahan sesuai dengan perkembangan jaman harus di antisipasi juga dari segi teknologi arsitektur dan teknologi aplikasinya.
2. Perancangan WIS aplikasi ini untuk ke depannya nanti dapat di kembangkan baik dari segi main aplikasi atau tenant aplikasi. Pada main aplikasi sebaiknya adanya tambahan rancangan untuk mengakomodasi satu tenant mempunyai 2 aplikasi tenant WIS berbeda (contoh aplikasi WIS ini dan aplikasi POS) secara terintegrasi apabila di mana ada satu kasus bahwa itu di butuhkan, sedangkan untuk aplikasi tenant tentu ini masih banyak yang dapat di kembangkan lagi seperti penambahan fitur rak , package serta dari *warehouse* lain nya, perhitungan harga item berdasarkan stok *average*, serta penambahan aplikasi seperti POS (point of sale), akuntansi dan lain nya.

DAFTAR PUSTAKA

[1] S. Machmud dan I. Sidharta, “Model Kajian Pendekatan Manajemen Strategik Dalam Peningkatan Sektor UMKM Di Kota Bandung (Model Study of Strategic Management Approach In SMEs Sector Improvement In Bandung),” *Jurnal Computech & Bisnis*, vol. 7, no. 1, pp. 56-66, 2013.

[2] M. Hompel dan T. Schmidt, *Warehouse management: automation and organisation of*

- warehouse and order picking systems, vol. 8, Springer Science & Business Media, 2006, pp. 1-7.
- [3] A. Dennis, B. H. Wixom dan R. M. Roth, *Systems analysis and design*, John wiley & sons, 2008.
- [4] Y. Septiana, “Perencanaan Strategis Sistem Informasi Dengan Pendekatan Ward and Peppard Model (Studi Kasus: Klinik INTI Garut),” *Jurnal Wawasan Ilmiah*, vol. 8, no. 1, 2017.
- [5] A. Mulyani, “Perencanaan Strategis Sistem Informasi Taman Satwa Menggunakan Metodologi Ward and Peppard,” *Jurnal Algoritma*, vol. 14, no. 1, 2017.
- [6] A. Mulyani dan D. Kurniadi, “Analisis Penerimaan Teknologi Student Information Terminal (S-IT) Dengan Menggunakan Technology Acceptance Model (TAM),” *Jurnal Wawasan Ilmiah*, vol. 7, no. 12, 2015.
- [7] S. Genta dan F. A. P. Utomo, “Warehouse Management System,” *Information System Application*, vol. 1, no. 1, 2016.
- [8] T. Kristanti dan N. Pamela, “Penerapan Knowledge Management System Berbasis Website CMS pada Divisi Produksi CV. Indotai Pratama Jaya,” *Jurnal Sistem Informasi*, vol. 6, no. 1, 2012.
- [9] M. Kara, “Kontribusi Pembiayaan Perbankan Syariah Terhadap Pengembangan Usaha Mikro, Kecil, Dan Menengah,” *AHKAM: Jurnal Ilmu Syariah*, vol. 13, no. 2, 2013.
- [10] A. S. Hidayat dan R. W. Witjaksono, “Perancangan Sistem Erp Dengan Modul Purchasing Dan Inventory Berbasis Odoo 9 Dengan Metode Asap Pada Pt. Aretha Nusantara Farm,” *eProceedings of Engineering*, vol. 3, no. 2, 2017.
- [11] F. Mubarak, H. Harliana dan I. Hadijah, “Perbandingan Antara Metode RUP dan Prototype Dalam Aplikasi Penerimaan Siswa Baru Berbasis Web,” *Creative Information Technology Journal*, vol. 2, no. 2, pp. 114-127, 2015.
- [12] K. Scott, *The unified process explained*, Addison-Wesley Longman Publishing Co., Inc., 2002.
- [13] Y. Septiana, D. Kurniadi dan A. Mulyani, “Perancangan Program Aplikasi Faraidh sebagai Sistem Pendukung Keputusan Pembagian Harta Waris Berorientasi Solver,” *Jurnal Algoritma*, vol. 14, no. 2, pp. 474-480, 2017.
- [14] Y. Liang dan F. Zhou, “A two-warehouse inventory model for deteriorating items under conditionally permissible delay in payment,” *Applied Mathematical Modelling*, vol. 35, no. 5, pp. 2221-2231, 2011.
- [15] D. Kurniadi, S. Sasmoko, H. L. H. S. Warnars dan F. L. Gaol, “Software size measurement of student information terminal with use case point,” *In Cybernetics and Computational Intelligence (CyberneticsCom), 2017 IEEE International Conference*, pp. 164-169, 2017.
- [16] Y. Septiana, “Design of prototype decision support system for flood detection based on ultrasonic sensor,” *MATEC Web of Conferences*, vol. 197, p. 03017, 2018.
- [17] Y. Septiana, D. Kurniadi, A. Mulyani dan W. Baswardono, “Design of decision support system for blood analysis,” *MATEC Web of Conferences*, vol. 197, p. 03018, 2018.
- [18] P. Sulistyorini, “Pemodelan Visual dengan Menggunakan UML dan Rational Rose,” *Dinamik-Jurnal Teknologi Informasi*, vol. 14, no. 1, 2009.
- [19] A. Dennis, B. H. Wixom dan D. Tegarden, *Systems analysis and design: An object-oriented approach with UML*, John wiley & sons, 2015.
- [20] P. Sulistyorini, “Pemodelan Visual dengan Menggunakan UML dan Rational Rose,” *Dinamik-Jurnal Teknologi Informasi*, vol. 14, no. 1, 2009.