

Manajemen Resiko dalam Pengembangan Aplikasi Berbasis *Agile Scrum* (Studi Kasus : Proyek ERP CV. Shankara Prima Indonesia)

Risk Management in Agile Scrum Based Software Development (Case Study : ERP Project CV. Shankara Prima Indonesia)

1st Muhammad Dzaky Hidayat
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia

thedzakyhidayat@student.telkomuniversity.ac.id

2nd Dana Sulistiyo Kusumo
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia

danakusumo@telkomuniversity.ac.id

Abstrak—CV Shankara Prima Indonesia adalah UMKM bergerak dibidang eksportir produk pertanian nusantara, untuk meningkatkan kredibilitas dan efektifitas bisnis usaha maka perusahaan membuat sistem menggunakan openERP yang terdiri dari modul website, CRM dan Inventory. Agar Infrastrukturnya cepat dirampungkan tim menggunakan pengembangan metode *Agile Scrum*, akan tetapi terdapat indikasi dan potensi aktifitas risiko yang dapat menyebabkan kegagalan proyek. Maka diajukan integrasi manajemen risiko dengan metode *Risk Poker* kedalam proyek Agile. Proses Assesment menggunakan Kanban risklog yang dilandasi sprint backlog, analisis dan mitigasi dilakukan bersamaan dengan fase sprint sehingga tidak mengganggu siklus Agile dan produk tetap dapat di deliver. Dari hasil evaluasi aktifitas manajemen risiko didalam proyek pengembangan openERP dan metode yang diterapkan didapatkan hasil tingkat efektifitas penanganan 0.2941, 0.7667 dan 1 untuk 3 sprint berturut turut dengan insiden paling banyak terjadi di kategori development, operational dan user dari pengamatan lapangan metode yang diusulkan tidak mengganggu siklus Agile namun insiden terdapat 9 insiden risiko yang berasal dari luar tasklist tidak berhasil diawasi oleh metode. Jumlah insiden risiko dan nilai komulatif risk point berangsur turun seiring sprint 2 karena tim dan sistem yang dibuat sudah stabil.

Kata kunci—*agile*, manajemen resiko, *risk poker*, *assessment* risiko

Abstract—CV Shankara Prima Indonesia is a small bussiness engaged in exporters of raw commodities for agricultural products from across Indonesia, to improve business effectiveness and effectiveness, the company makes a system using openERP which consists of a website module, CRM and Inventory. In order for the

infrastructure to be quickly completed company use Agile development methods, however there are indications and potential risks that can lead to project failure. Therefore, it is proposed to integrate risk management with the Risk Poker method into the Agile development. The assessment process uses the Kanban risklog based on the sprint backlog, analysis and mitigation are carried out simultaneously with the sprint phase so that it does not disrupt the Agile cycle and the product can still be delivered. From the results of the evaluation of risk management activity of the openERP development project and the methods applied in it, the results showed that the level of effectiveness of handling 0.2941, 0.7667 and 1 for 3 consecutive sprints and the most incidents occurred within development, operational, and user risk categories and from the observations the method did not disrupt the Agile cycle but there were 9 unrecorded risk incidents that came from outside the tasklist unsupervised. The number of risk incidents and the cumulative value of risk points decreased with sprint 2 because the team and the system considered stable.

Keywords—*agile*, risk management, *risk poker*, risk assessment

I. PENDAHULUAN

Pada bagian pendahuluan terdiri dari sub bahasan Latar belakang, Topik dan batasannya dan Organisasi Penulisan, secara rinci tiap bahasan dibahas sebagai berikut.

A. Latar Belakang

Saat ini siklus pengembangan *Agile* banyak digunakan oleh perusahaan rintisan dan umkm, Secara umum, agile method adalah proses menerapkan siklus pendek berulang, aktif

melibatkan pengguna untuk membangun, memprioritaskan, memverifikasi kebutuhan, dan bergantung pada pengetahuan teknis tim menggantikan dokumentasi [1]. Metodologi *Agile* cenderung mengelola risiko proyek secara implisit[17][16]. Dengan proses manajemen risiko eksplisit yang ditambahkan ke metode *Agile*, ada *trade-off* dengan lebih banyak artefak (misalnya log risiko bersama dengan backlog produk) dan proses yang panjang termasuk identifikasi risiko, penilaian, prioritas, dan mitigasi justru menambah beban pengembangan *Agile*[3].

Studi kasus menggunakan subjek CV Shankara Prima Indonesia, produk usaha perusahaan ini ialah eksportir komoditas produk tani nusantara, untuk mendukung proses bisnisnya perusahaan tersebut menginginkan aplikasi ERP (*Enterprise Resource Planning*) dan CRM (*customer relationship Management*). penerapan ERP dapat meningkatkan kemampuan unit bisnis untuk mengelola sumber dayanya secara efektif [4] CV Shankara Prima pun menggunakan ERP ini sebagai infrastruktur untuk ditunjukkan pada client, meningkatkan kredibilitas perusahaan walaupun masih berskala kecil.

Dari kondisi yang telah dijelaskan proyek pengembangan ERP CV shankara cocok menggunakan siklus pengembangan *Agile* karena memiliki kriteria mengacu pada studi [13] yakni:

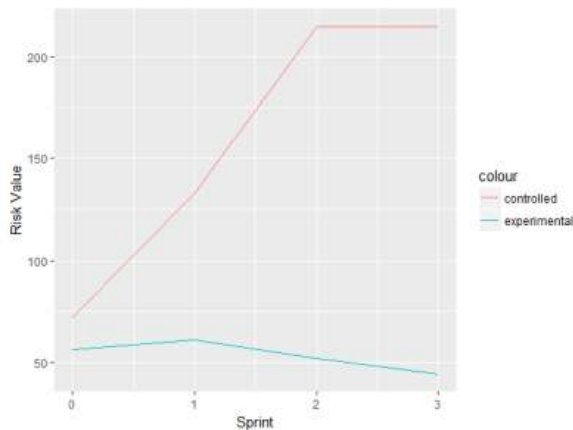
1. Aplikasi terdiri dari modul-modul independent
2. Menurut User proyeksi pengembangan aplikasi bersifat
3. Fitur bergantung oleh *user story* dan value.

4. *User* menginginkan produk yang cepat dapat digunakan.

Akan tetapi pada siklus sprint awal proyek pengembangan aplikasi *Agile* akan mengalami tingkat risiko tertinggi [5][16][17] dan studi menyatakan bahwa 34% dari proyek berbasis integrasi manajemen IT mengalami kegagalan dan pengembangan produk IT baru mengalami kegagalan sebanyak 12% [14]. Selain itu diidentifikasi ada kriteria hambatan yang dialami tim pengembang yang berpotensi dapat mengurangi efektifitas siklus *Agile*[15][17] yakni:

1. Tim pengembang ERP terdiri dari pemula
2. Pengembang bukan merupakan tim tetap
3. Ada batasan pendanaan
4. Kolaborasi dengan *customer* dan *end user* terhalang bahasa, kultur dan jarak.

Dibanding metode pengembangan *Agile* lainnya Scrum dipakai sebanyak 58% proyek, Diikuti oleh metode Kanban 7% dan Extreme Programming 8%, sisanya adalah metode campuran[23] Dalam metode scrum tiap fase pengembangan dipecah dalam skala kecil dieksekusi menyesuaikan tiap langkah peningkatan produk kedalam iterasi yang disebut Sprint[23] Dalam siklus Scrum *Agile* secara garis besar hanya mengidentifikasi tanpa ada langkah analisis dan manajemen risikonya [17]. Namun dengan adanya manajemen risiko yang dapat menurunkan kondisi ketidakpastian dan meningkatkan kesuksesan proyek aplikasi[17] Studi terkait menjelaskan bahwa terjadi eskalasi dari risk value apabila dari tiap fase sprint risiko tidak segera ditangani.



GAMBAR 1
GRAFIK PERBANDINGAN GROSS RISK VALUE ANTARA SCRUM DENGAN MANAJEMEN RISIKO(BIRU) TERHADAP SCRUM TANPA MANAJEMEN RISIKO[5]

Dari grafik terlihat insiden risiko yang tidak segera ditangani di dalam bahasan studi akan menimbulkan nilai risk value terus bertambah, berbeda jika di tiap siklus ditangani insiden risiko cenderung turun, karena tidak ada tugas penanganan turunan dari sprint sebelumnya[5]. Manajemen risiko dilakukan dalam timeframe pendek, tiap risk yang teregister dievaluasi pada tiap Sprint *planning meeting*, meeting harian dan meeting evaluasi

penutup iterasi sprint, dengan tipe langkah manajemen risiko menyesuaikan. Sedangkan pada metode Waterfall perencanaan mitigasi risiko dilakukan paling awal saat pembukaan proyek, langkahnya dilakukan dengan cara mengestimasi risiko yang akan bermunculan nantinya dan terus dimonitor dan di kendalikan saat muncul[24]

Dari sisi proyek pengembangan ERP CV shankara ini pun manajemen risiko dapat mendukung

ekosistem dari tim yang dapat saling memonitor walau terdapat batasan kultur dan jarak dan dapat mendukung fungsi substansinya [14].

Metode yang diusulkan dalam pengembangan proyek ERP CV Shankara Prima Indonesia ialah Risk Poker karena pengukuran dan kompleksitas risiko dapat diambil dari user story, *tasklist* dan dapat dipasang di setiap sprint[18], dibanding dengan menggunakan metode DSDM yang umum untuk manajemen risiko proyek perlu terlebih dahulu membuat risklog berisi jenis risiko umum dan potensi risiko yang sering teregister sesuai pedoman konsorsium dan jadwal evaluasi terpisah dari siklus sprint [19]dengan pemilihan metode risk poker dan metodenya tersebut tidak mengganggu kaidah siklus Agile yang dituntut untuk cepat dan dapat diintegrasikan didalam lingkungan proyek openERP CV Shankara Prima Indonesia.

B. Topik dan Batasannya

Berdasarkan latar belakang yang disajikan, penulis akan membangun sistem pengukuran, analisis mitigasi, dan evaluasi efektifitas manajemen risiko menggunakan metode Risk Poker. Dengan metode risk poker maka siklus manajemen risiko tergabung dengan komponen siklus agile Scrum, Kemudian output akhirnya dapat menghasilkan metrik untuk menjawab bagaimana efektifitas penanganan insiden risiko yang terjadi dari hasil proyek pengembangan ERP CV Shankara Prima Indonesia sesuai metode yang dipilih.

Adapun batasan bahasan penelitian ini pada sprint 0 sampai sprint 2 karena pada fase ini insiden risiko cenderung lebih penting diawasi karena potensi risiko lebih sering terjadi [5][16][17]

C. Tujuan

Tujuan yang ditetapkan oleh penulis untuk menjadi output tugas akhir ini ialah bagaimana membuat sistem manajemen risiko menggunakan metode risk poker yang terdiri dari assessment, penanganan dan evaluasi dalam lingkungan proyek pengembangan openERP berbasis Agile di CV Shankara Prima Indonesia kemudian

mendokumentasikan langkah penilaian, penanganan dan pengukuran efektifitas dari manajemen risiko yang dilakukan.

D. Organisasi Tulisan

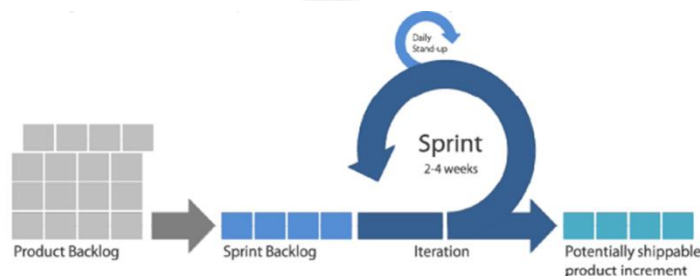
Komposisi penulisan penelitian terdiri dari bagian 1 pendahuluan menjelaskan garis besar penelitian, bagian 2 studi berisi rangkuman sumber pustaka yang dipakai, pada bagian 3 perancangan merupakan desain dirancang untuk menyelesaikan permasalahan penelitian, dan bagian 4 evaluasi berisi pengukuran efektifitas hasil rancangan ditutup dengan bagian 5 yang merupakan kesimpulan dari hasil penelitian.

II. KAJIAN TEORI

A. Agile Scrum Software Development Methods

Agile software development methods atau *agile methodology* merupakan sekumpulan metodologi pengembangan perangkat lunak yang berbasis pada pengembangan iteratif, di mana persyaratan dan solusi berkembang melalui kolaborasi antar tim yang terorganisir (Pressman, 2010). Sementara Sommerville (2011) mengemukakan metode agile merupakan metode pengembangan *incremental* yang fokus pada perkembangan yang cepat, perangkat lunak yang dirilis bertahap, mengurangi *overhead* proses, dan menghasilkan kode berkualitas tinggi dan pada proses perkembangannya melibatkan pelanggan secara langsung. Ada beberapa model pengembangan perangkat lunak yang termasuk agile software development methods, yaitu 1) *Extreme Programming*, 2) *Adaptive Software Development*, 3) *Dynamic Systems Development Method*, 4) *Model Scrum*, dan 5) *Agile Modeling*. Di dalam penelitian ini, model yang akan digunakan adalah model scrum.

Menurut Pressman (2010) model scrum adalah metode pengembangan peranti lunak secara cepat (agile). Prinsip scrum sesuai dengan prinsip-prinsip yang terdapat pada metode pengembangan peranti secara cepat yang digunakan untuk menuntun kegiatan pengembangan peranti lunak, seperti pemenuhan kebutuhan, analisa, desain, dan penyampaian (delivery).



GAMBAR 2
FRAMEWORK SCRUM AGILE[5]

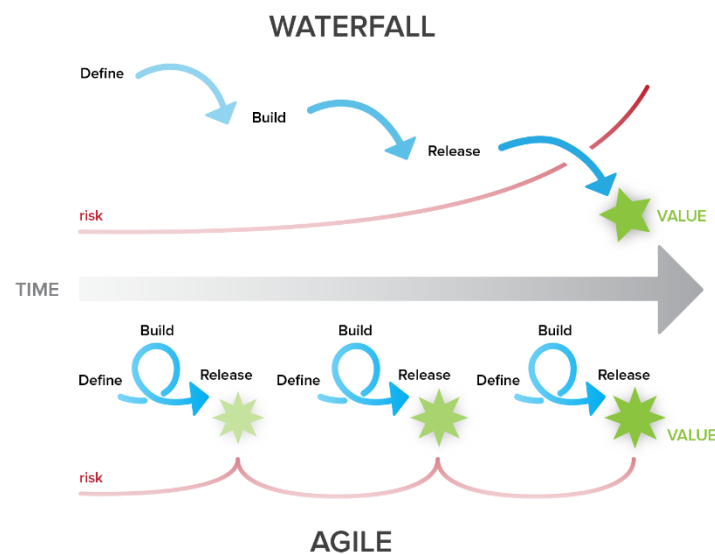
Scrum terdiri dari iterasi yang disebut sprint, di fase planning terdapat sprint backlog yang berfungsi

untuk memperbaiki tujuan pada iterasi sprint yang sedang dilakukan, pada fase planning di sprint 0

sebelumnya, terdapat Product backlog berisi cerita kebutuhan dari user yang belum dipetakan secara spesifik. Pada masa iterasi adalah proses pemrograman, proses problem solving dan daily meeting, masa ini dilakukan dalam rentang 2-4 minggu per fase sprint. Diakhir masa sprint adalah Sprint retrospective, aktifitas ini berfungsi untuk melihat apa yang sudah dikerjakan, lalu kemudian berdiskusi untuk langkah peningkatan, hasil diskusi ini menjadi laporan evaluasi untuk sprint saat ini sekaligus sprint backlog untuk sprint selanjutnya.[26]

Risiko yang paling umum terjadi pada aspek komunikasi, pembelajaran dokumentasi dan perilaku individu tim[17], dan dalam pengembangan aplikasi aspek penyumbang risiko terbesar berhubungan dengan Batasan waktu, komunikasi tim, tool dan staf, banyak proyek gagal karena tidak memberikan langkah penanganan khusus apabila terjadi insiden risiko[27].

Dibanding dengan metode waterfall yang saat ini penggunaannya sepertiga dari metode Agile[23], konsistensi nilai dan kejadian risiko cenderung turun seiring berjalannya proyek[20] seperti terlihat pada gambar 3.



GAMBAR 3
PERBANDINGAN NILAI RISIKO AGILE VS WATERFALL.[C]

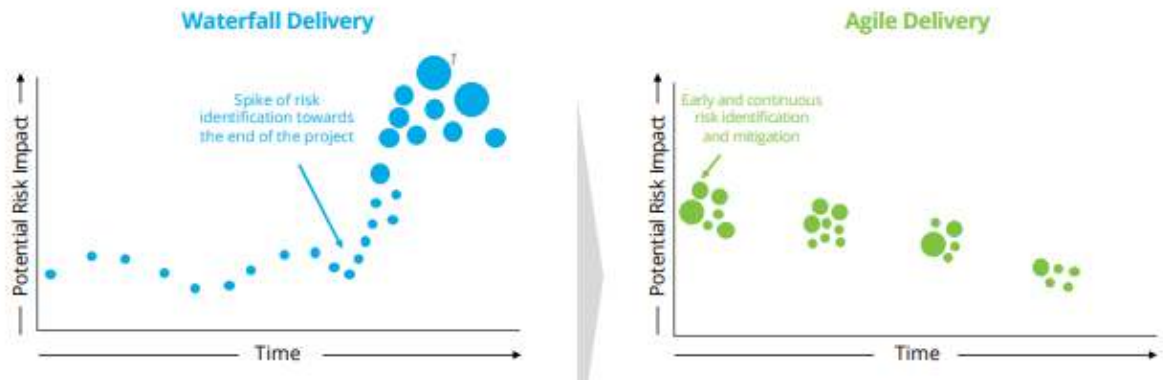
Pada penanganan risiko di metode waterfall terdapat beberapa faktor yang menunjang tingginya risiko kumulatif di akhir proyek[28]

1. Secara umum Waterfall mengalir turun. Apabila ada isu kritis maka harus kembali ke langkah sebelumnya menyebabkan delay dan penurunan utilitas tim
2. Tidak ada kerja parallel dalam tim
3. Ketergabungan customer minim sehingga spesifikasi produk berpotensi dibawah ekspektasi customer

Risiko pada metode Waterfall ditangani oleh tim, tools, dan *project charter* sesuai panduan risk

management sesuai ISO. *Project charter/risk charter* diawali dengan manager mengikuti perkembangan risiko yang terdeteksi, apabila risiko terus tumbuh maka developer dan manajer mencoba menemukan fitur yang menguatkan risiko kemudian menggantinya hingga level risiko turun.

Akan tetapi nilai risiko untuk agile cenderung tinggi diawal fase pengembangan proyek[20]. Potensi risiko dan insidennya terjadi di tiap sprint, akan tetapi nilai risiko tidak mengalami kenaikan karena tiap siklus memiliki penanganannya sendiri[21] seperti terlihat pada Gambar 4.



GAMBAR 4

PERBANDINGAN BESARAN NILAI RISIKO DAN RISK IMPACT ANTARA PENGEMBANGAN METODE WATERFALL DAN AGILE.[21]

Dari Gambar 4 terlihat bahwa risiko yang ditangani tiap sprint tidak dieskalasi ke sprint selanjutnya, akan tetapi pada metode waterfall dieskalasi hingga mencapai puncak pada akhir rilis produk aplikasi[21]. Secara umum jenis risiko yang sering terjadi pada pengembangan berbasis Agile dari segi finansial, teknikal dan proses bisnis[22], secara penanganannya tim di siklus metode Agile Scrum fokus mengimplementasikan kebutuhan produk sesuai yang dicantumkan pada fase planning, alasan mengapa risiko Agile rendah saat di akhir proyek seperti pada gambar 4, dikarenakan kebutuhan spesifikasi terus disajikan tiap fase Sprint sehingga customer dapat aktif memberi feedback[23].

Berbagai studi mengenai manajemen risiko dalam proyek Agile dirangkum dan didapatkan 3 metode yang diusulkan oleh penelitian lampau[5][6][7][8], berbagai macam metode yang dimaksud ialah *Dynamic system development Method*, *Risk Based Scrum Method* dan *Risk Poker technique* Teknik risk poker yang digunakan berasal dari teknik planning poker. Hasil evaluasi menunjukkan bahwa teknik *risk poker* lebih baik daripada teknik lainnya[6][9] Teknik poker adalah Teknik estimasi risiko yang dilakukan diawal sprint dengan menggunakan user story dan story point sebagai parameter ukuran dasar untuk potensi risiko yang terjadi di tiap fase[18] Semua risiko utama terdaftar dalam matriks dan semua risiko dipantau terus menerus selama seluruh proses pengembangan

B. Studi terkait



GAMBAR 5

SISTEM KANBAN PENCATATAN PRIORITAS RISK [9]

Untuk mengkonversi parameter kualitatif resiko kedalam ukuran metrik menggunakan point Risk[11], risk point digunakan untuk mengukur risk kumulatif didalam proyek

$$Risk\ Point = PCF \times URPW \text{ (persamaan 1)}$$

PCF adalah *Project Characteristic Factor* dan URPW adalah *Unadjusted Risk Point Weight* PCF adalah nilai untuk memberikan bobot pada proyek dan menyesuaikan nilai akhir metrik berdasarkan

faktor teknis dan lingkungan pengembangan, PCF dirumuskan melalui model matematika yang dikembangkan sesuai proyek, dalam penelitian ini merujuk pada paper penelitian [5]

$$PCF = 1,05 + (0,015 \times CF) \text{ (persamaan 2)}$$

CF adalah *characteristic factor* ditentukan dari n pertanyaan di kali bobot antara 0 sampai 4, berikut persamaannya:

$$CF = \sum^n (pertanyaan\ i \times\ bobot) \text{ (persamaan 3)}$$

URPW ditentukan dari akumulasi dari bobot tiap resiko teridentifikasi sesuai nilai table berikut:

TABEL 1
KONVERSI RE KE BOBOT URPW[11]

Classification	RE(Risk)	Weight(Risk)
Very Low	[0,0, 0,2)	1
Low	[0,2, 0,4)	2
Average	[0,4, 0,6)	3
High	[0,6, 0,8)	4
Very High	[0,8, 1,0]	5

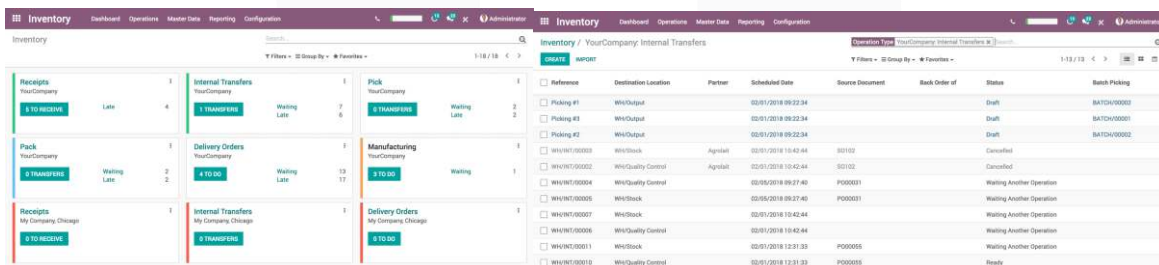
RE atau Risk Exposure diestimasi dari persentase kemungkinan terjadinya resiko dikali dengan Impact resiko. Nilai Risk Point akhir dipakai untuk menentukan prioritas, kebijakan mitigasi dan mengukur secara kasar kualitas produk tiap sprint.

C. Deskripsi Subjek CV Shankara Prima Indonesia

Usaha dirintis sejak tahun 2021, dengan badan usaha berfokus dibidang ekspor komoditi pertanian, di Garut dengan koperasi desa setempat sebagai mitra perusahaan sedang aktif mengelola komoditas kopi dan produk palawija dengan kapasitas produksi 20 ton perbulan. *Workflow* proses bisnis dimulai dengan mencari *supplier* ke koperasi desa kemudian membuat kontrak kerjasama kemitraan koperasi, secara aktif perusahaan mencatat kapasitas minguang koperasi desa sembari mencari calon buyer di marketplace dan komunitas, perusahaan

juga membuat direktori kontak potensial buyer dan validasi identitas histori buyer di kemendag. Proses jual beli diawali dengan komunikasi dan diskusi harga, fase kontrak jual beli secara resmi setelah rilis *offering letter*, *letter of intent*, *full corporate offer*, *procurement invoice*, *sales contract* dan *commercial invoice*. Transaksi keuangan oleh bank dengan skema transaksi antar negara diikat *letter of credit*. Proses pengiriman komoditas ditangani oleh jasa *forwarder* untuk pengurusan *packaging* dan *cargo*

tim pengembang mengusulkan penggunaan SDK (*Software Development Kit*) *opensource* Odoo sebagai aplikasi dasarnya, Odoo SDK memiliki fitur pengembangan *webapp* berbasis MVC (*Model View Controller*) dengan modul ERP dan CRM berstandar industri sesuai kebutuhan niaga perusahaan[12], database ditangani menggunakan PostgreSQL.



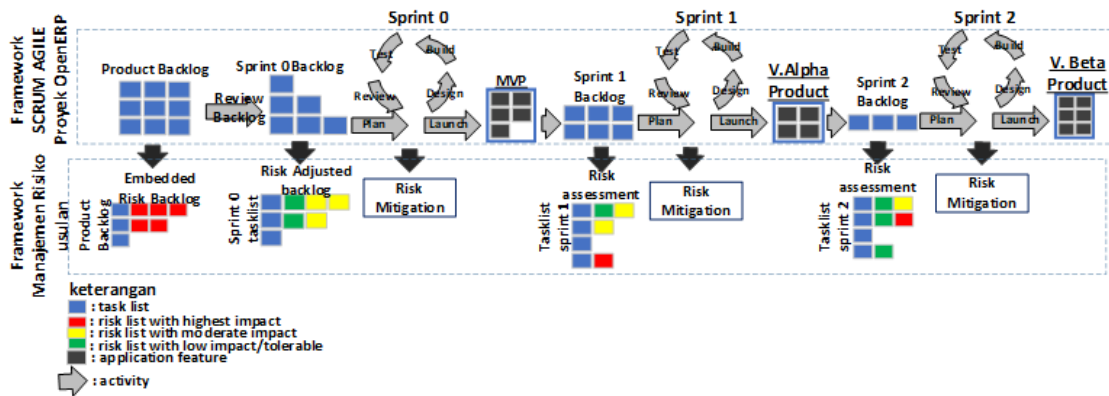
GAMBAR 6
ODOO MODUL INVENTORY [12]

III. METODE

Dibagian ini dibahas detail mengenai siklus Agile yang digunakan dalam pengembangan proyek sehingga menghasilkan produk yang akan ditampilkan di bagian 4, bersama dengan metode manajemen risiko yang diawali dengan metode assessment, metode analisis dan miitigasi risikonya bersama ukuran metode evaluasi keberhasilan manajemen risiko yang telah dirancang.

A. Deskripsi Umum Sistem

Alur sistem yang dibangun mengikuti *framework* agile pembangunan ERP CV Shankara Prima Indonesia sampai dengan digabungkan metode manajemen risiko menggunakan Risk Poker. Berikut ini menjelaskan bagaimana aktivitas yang dilakukan oleh tim didalam siklus Agile, kemudian pada bahasan selanjutnya mengenai penerapan manajemen risiko kedalam siklus Agile yang dibuat.



GAMBAR 7

INTEGRASI FRAMEWORK AGILE/SCRUM PADA PROYEK PEMBUATAN OPENERP CV SHANKARA DENGAN FRAMEWORK MANAJEMEN RISIKO USULAN

Siklus *Agile* mengikuti pola umum proses pengembangan *Agile* seperti yang tertera dalam bagian 2.1 dan gambar 2. Dimulai dari Product Backlog hingga pengembangan versi Alpha. Siklus

Agile dimulai dengan mengumpulkan Product Backlog yang berasal dari user story yang dikemukakan oleh pemilik perusahaan sendiri.

TABEL 2
PRODUCT BACKLOG VERSI 0

No.	User Stories	Priority
1	Punya fitur pendataan inventori lengkap	High
2	Bisa digunakan untuk orang tua dikoperasi	High
3	Penyimpanan profil buyer dan jasa forwarder	Medium
4	Website menampilkan komoditas desa	High
5	Website menampilkan informasi detail proses bisnis	High

Dari backlog diketahui user membutuhkan modul inventori, CRM dan website. Dari user story ditranslasi menjadi tasklist yang dipecah sesuai

modul aplikasi untuk dibagikan ke setiap anggota tim

Product Backlog	TaskList		
A. Punya fitur pendataan inventori lengkap	A.1 CRUD Inventori	A.2 CRUD stock	A.3 CRUD entitas koperasi
B. Bisa digunakan untuk orang tua di koperasi	B.1 utilisasi dropdown/click input	B.2 maximum input activity to 5 input	
C. Penyimpanan profil buyer dan jasa forwarder	C.1 CRUD entitas buyer & forwarder		
D. Website menampilkan komoditas desa	D.1 blok profil komoditas komunitas		
E. Website menampilkan informasi detail proses bisnis	E.1 blok profil bisnis dan kontak	E.2 blok portofolio	

GAMBAR 8

REKAP TASKLIST PROYEK APLIKASI OPENERP DAN WEBSITE

Menurut pendiri perusahaan, hal hal tersebut adalah kebutuhan dasar yang harus ada dari aplikasi untuk proses bisnis, dikatakan bahwa penambahan dan perubahan fitur melihat terlebih dahulu produk versi alpha *minimum viable product*, lalu dilanjutkan sesuai kebutuhan koperasi mitra.

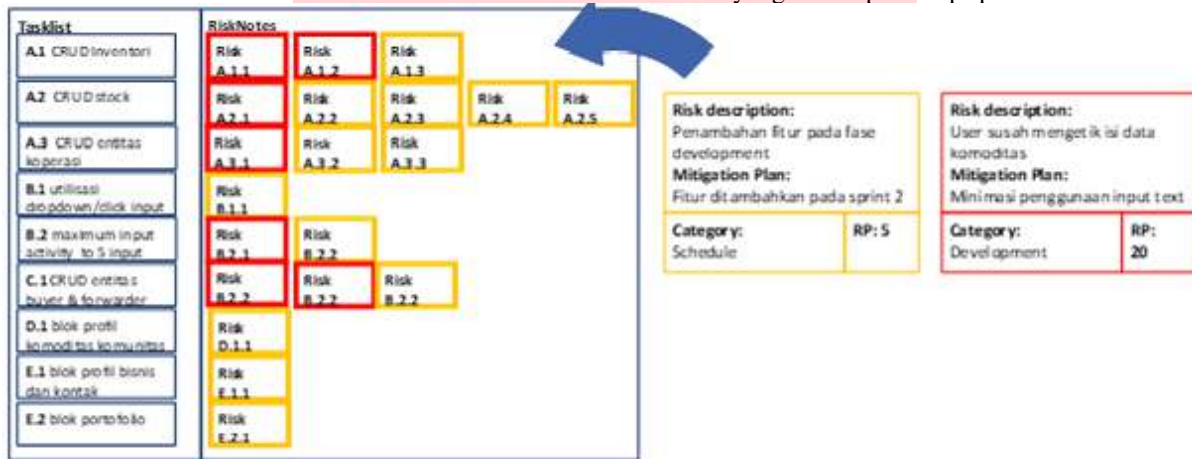
Dari backlog yang sudah dikeluarkan, maka tim akan mengembangkan produk berdasarkan backlog tersebut, tahap sprint terdiri dari *planning, design, build test review dan launch product*. Hasil pengembangan proyek berbasis Agile akan disajikan di awal bagian 4 menampilkan hasil produk openERP yakni website, modul CRM dan modul Inventory.

Framework manajemen risiko yang digabungkan pada siklus membutuhkan backlog seperti siklus Agile Scrum yang dipakai. Proses diawali dengan membuat *Embedded Risk Backlog*, Selanjutnya *Embedded Risk backlog* dirumuskan menjadi *Risk Adjusted backlog* didalam backlog ini

risiko dari fase sebelumnya sudah dipilah, dianalisis dan disortir menyesuaikan Sprint backlog sprint 0. Dari Risk Adjusted Backlog kemudian dimitigasi apabila insiden tertulis atau lainnya muncul seiring pengembangan produk. Ketika produk terbit dari Sprint 0 maka risk backlog dalam risk evaluation ikut di evaluasi. Ketika feature backlog sudah berubah menjadi sprint backlog untuk persiapan di Sprint 1, maka review Risk backlog ikut menyesuaikan begitu siklusnya hingga sprint 2. Arti penanda warna dan model Kanban risk backlog dalam penelitian ini dijelaskan pada point tool Assesmen Risiko di bagian 3.2. dalam penelitian ini Kanban Risk Backlog menggunakan Aplikasi Web Kanban yang dilampirkan..

B. Assessment Risiko

Scrum master bersama tim membuat dan merumuskan potensi risiko yang mungkin terjadi berdasarkan *User story* pada table 2 dan sesuai tasklist yang dibuat pada tiap sprint.



GAMBAR 9 SISTEM KANBAN RISK LOG (KIRI) DAN DETAIL ISI RISK NOTE/LOG YANG TERCATAT (KANAN)

Sesuai metode risk poker pada gambar 5. Assessment menghasilkan Kanban dengan kode warna sesuai prioritasnya, risk point kritis mendapat kode merah(prioritas) artinya belum ditangani atau memiliki efek besar, kuning untuk prioritas *moderate* dan warna hijau apabila risiko sudah ditoleransi atau diselesaikan. Tiap risk memiliki kode yang berasal dari turunan nomor Tasklist, apabila risiko tidak terdaftar muncul maka kode penomoran dinamai dengan kode “SP” diikuti dengan nomor sprint.

pada *tool assessment* dengan bantuan kuantifikasi risk point, risk point didapatkan dari persamaan berikut

$$Risk\ point = 2 \times Risk\ Level - Effort\ Level.$$

(persamaan 4)[6]

risiko yang muncul dapat dikalkulasi dampaknya dan penanganannya secara objektif meminimasi waktu diskusi dan perdebatan antar anggota tim, dalam kolom kategori pada gambar digunakan untuk mengelompokkan asal risiko dan departemennya ditandai oleh deskripsi kategori dan nomor asal tasklist.

C. Analisis dan Mitigasi Risiko

Fase penanganan dirancang untuk efisiensi waktu, dengan risk poker menghasilkan flow penanganan yang diringkas tidak mengganggu proses sprint dengan menggabungkan dengan siklus sprint tanpa menambahkan aktivitas lain, metode analisis dan penanganan mengikuti flow berikut



GAMBAR 10 ALUR POLA PENANGANAN MANAJEMEN RISIKO PROYEK

Di fase sprint risiko yang perlu segera dimitigasi ialah teknikal *development*, fase awal dimulai dari *risk point* dan kategori risikonya, jika tidak kritis maka ditoleransi, namun jika sifatnya kritis tetapi bukan disisi *development* maka dikonsultasikan dengan user untuk penanganannya, apabila kategori *development* maka penyelesaian ditangani oleh tim developer. Prioritas penanganan juga memperhatikan nilai risk point sesuai strategi metode Risk Poker[6]

IF Threat Score ≥ 14 THEN Solve threat, ELSE Ignore threat [6]

Framework analisis dan mitigasi dipasang pada tiap fase sprint, Teknik penanganan mitigasi dalam proyek ini adalah penghindaran, retensi, dibagi, transfer, pencegahan, toleransi dan reduksi. Kemudian potensi tersebut diklasifikasi kembali statusnya. Apabila risiko telah mencapai note hijau risk pointnya dibawah ambang atau bernilai 0 maka insiden tidak perlu dieskalasi ke sprint selanjutnya.

D. Evaluasi efektifitas penanganan Risiko

Metrik disajikan untuk mengukur keberhasilan mitigasi yang diajukan pada bagian 3.3. Metrik ini menjadi acuan mengenai tingkat keberhasilan

metode yang dipakai dan sistem yang dirancang dilingkungan proyek pengembangan aplikasi ERP CV shankara, sebagai berikut:

$$Efektifitas = 1 - \left(\frac{risk\ point\ pasca\ mitigasi}{risk\ point\ pra\ mitigasi} \right) \text{ (persamaan 5)}$$

Nilai efektifitas dalam bentuk persentase dengan cara membandingkan risk point komulatif dari tiap risiko sebelum penanganan dan sesudah penanganan, dari sini dapat diketahui efektifitas penanganan tiap sprint, jenis risiko terberat dan siklus efektifitas selama tiap fase sprint. Manajemen risiko dikatakan berhasil apabila nilai efektifitas drastis mendekati angka 100% atau 1 selain dari nilai efektifitas peforma penanganan ditandai oleh lama waktu penanganan, banyak risiko yang dianulir dan rasio efektifitas antar sprint.

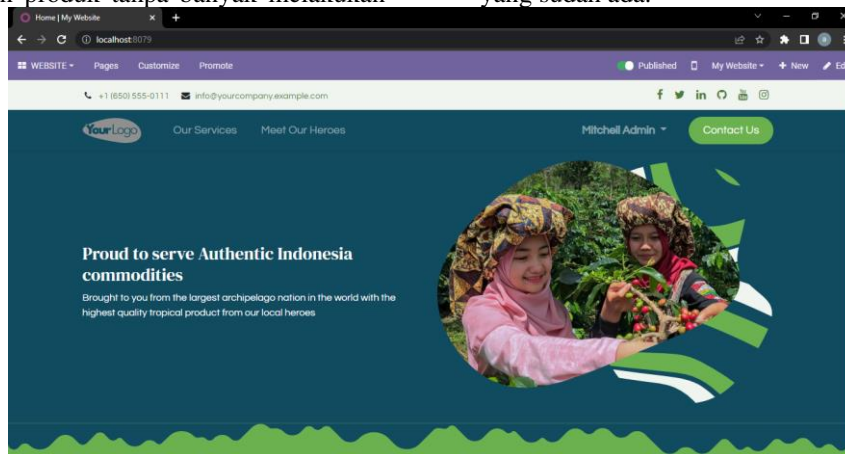
IV. HASIL DAN PEMBAHASAN

Pada bagian 4 ini dijelaskan menjadi 3 bagian, bagian 4.1 adalah hasil pengembangan produk kemudian pada bagian 4.2 membahas aktifitas penanganan sepanjang proses sprint 0 hingga sprint

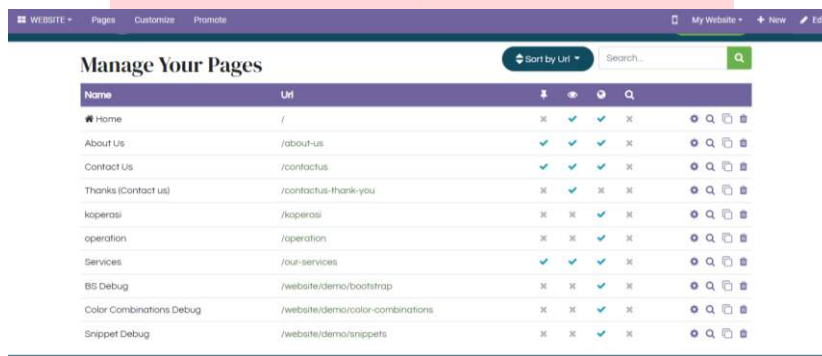
A. Hasil pengembangan produk

Dengan development Kit Odoo tim dapat mengembangkan produk tanpa banyak melakukan

aktivitas coding dan cukup memodifikasi modul yang sudah ada.



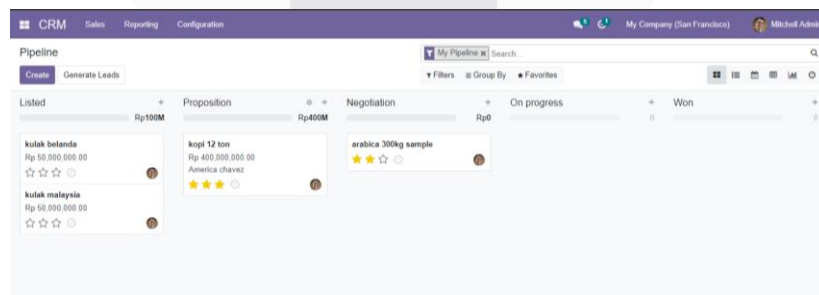
GAMBAR 11
TAMPILAN WEBSITE HASIL SPRINT 0



GAMBAR 12
ORGANISASI HALAMAN WEBSITE HASIL SPRINT 0

Modul web menggunakan asas 1 halaman dan 0 memakai prinsip *KISS (Keep It Simple, Stupid)* menampilkan semua informasi, fungsi utama dari halaman *website* sebagai bahan marketing dan menampilkan informasi singkat dan kredibel dari proses bisnis. Pada sprint 0 *website* menampilkan hal hal dasar yang telah ditentukan pada backlog,

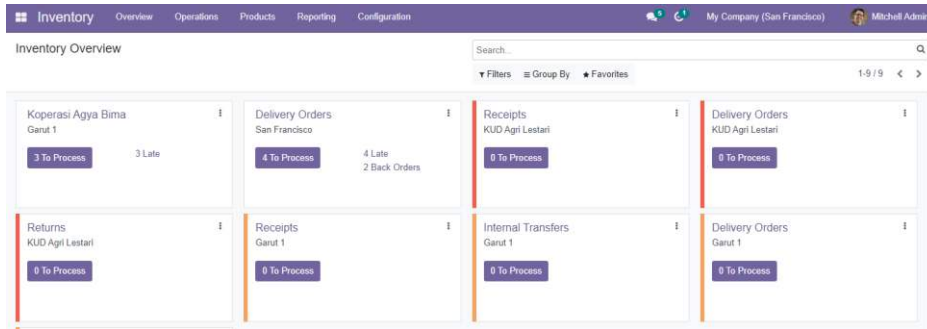
menampilkan profil badan usaha, komoditas, kontak, dan portofolio, pada sprint 1 user menginginkan ada fitur yang dapat memonitor secara aktif aktifitas transaksi dan inventori di koperasi. Organisasi halaman memperlihatkan halaman utama dan submenunya,



GAMBAR 13
TAMPILAN CRM HASIL SPRINT 0

Modul aplikasi CRM dalam sprint 0 bertujuan memberikan produk MVP (*Minimum viable product*) dan perkenalan tool *OpenERP Odoo* kepada user, menurut user modul CRM yang telah dimodifikasi basis kodenya ini cukup membantu dan

dapat dioperasikan dengan mudah, revisi modul ini akan dilanjutkan di fase sprint 1. Modul CRM dianggap memenuhi kebutuhan yang dibutuhkan user untuk mendaftarkan semua potensi transaksi yang berlangsung.



GAMBAR 14
TAMPILAN MODUL INVENTORI HASIL SPRINT 2

Modul inventori dikembangkan dengan memodifikasi kode dari modul yang sudah ada menyesuaikan *backlog*, hasil modul di sprint 2 masih mengalami penyesuaian setelah *user test* dan *review* namun menurut user modul inventori dianggap sudah MVP dan memenuhi ekspektasi fungsi dasarnya yakni menginventarisir dan mencatat sirkulasi bahan baku yang dikumpulkan mitra koperasi dari mitra petani.

Aktifitas tiap sprint dapat dilihat pada pembahasan berikut, tasklist 0 dari backlog user story sudah dilampirkan didalam gambar 8.

Pada sprint 1 perencanaan dan evaluasi *backlog* dibuat berdasarkan masing masing modul hasil dari pengembangan *backlog 0*. *Tasklist* tidak lagi merujuk pada user *backlog* seperti pada gambar 4, *tasklist* dari sprint 1 ditunjukkan dalam gambar 15.

Product Backlog sprint 1	Tasklist			
A. Modul website	A.1 penambahan peta lokasi usaha	A.2 penambahan detail katalog produk	A.3 user test	
B. Modul CRM	B.1 penambahan kolom log percakapan/catatan terakhir dengan klien	B.2 revisi info penawaran 1 harga klien menjadi info tentang harga tertinggi dan terendah	B.3 penambahan info nomor lain klem enperindag untuk setiap customer	B.4 user test
C. Modul Inventori	C.1 input inventori mitra menggunakan halaman web dihilangkan, pengguna mitra lebih paham memakai pesan teks telegram		C.2 user test	
D. Deployment dan testing	D.1 mempelajari terms and condition Deployment odoo			

GAMBAR 15
TASKLIST SPRINT 1

Dalam proses sprint 1 modul *website* user ingin menambahkan peta lokasi usaha dan detail katalog layaknya *e-commerce* produk umumnya, modul CRM mengalami perubahan dan penambahan fitur yakni penambahan kolom catatan percakapan terakhir dengan klien, sedangkan modul inventori dalam perombakan.

Hasil Sprint 1 dianggap sudah memenuhi dokumen *definition of ready SCRUM* dan ekspektasi user yang diajukan diawal kontrak proyek. Maka di fase Sprint 2 secara garis besar sistem versi beta siap di deploy secara publik, aktifitas yang tercatat dalam tasklist lebih pada fase deploy. *tasklist* sprint 2 dilampirkan pada Gambar 16

Product Backlog sprint 2	Tasklist		
A. Modul website	A.1 finalisasi publish demo	A.2 full integrasi	A.3 user test
B. Modul CRM	B.1 finalisasi publish demo	B.2 full integrasi	B.3 user test
C. Modul Inventori	C.1 finalisasi publish demo	C.2 full integrasi	C.3 user test
D. Deployment dan testing	D.1 deploy demo di alamat odoo.com		

GAMBAR 16
TASKLIST SPRINT 2

Pada sprint 2 ini aktifitas yang tercatat dalam tasklist lebih kepada integrasi dan finalisasi, karena produk dan ketiga modul yang dihasilkan sudah memenuhi ekspektasi user dan fungsionalitas.

B. Analisis Hasil Aktifitas Manajemen Risiko Proyek pengembangan

Hasil aktifitas manajemen risiko dari bagian 3 berupa Kanban di gambar 9 disajikan dalam bentuk rekap tabulasi note risk, berisi kategori risk, risk point dan langkah mitigasi, bentuk tabulasi disajikan untuk memudahkan penampilan data, secara rinci tampilan Kanban Risk Backlog dapat dilihat pada Lampiran 1.

TABEL 4
REKAPITULASI RISK NOTE EVALUASI SPRINT 0

kode tasklist	deskripsi	kategori	point risiko		mitigasi
			pra	pasca	
SP0.1	deploy demo dibatasi jam	dev	10	0	deploy demo local kemudian pitching ke user dan mitra
SP0.2	publisher tidak melayani publish container	devops	20	10	deploy demo local di sprint ini cari jasa hosting yang kapabel
SP0.2	publisher tidak melayani service postgre	devops	20	10	deploy demo local di sprint ini cari jasa hosting yang kapabel
SP0.3	development overtime	dev	30	30	revisi ke sprint selanjutnya
SP0.4	internet lama pulling image dependencies	dev	20	0	pindah ISP
SP0.5	user tidak bisa dihubungi/telat respon untuk risk action	operation	10	0	user menyanggupi fast respon
E.1.1	risiko vulnarebility web operasional dan web utama	operation	5	0	submenu operasi & koperasi dipisah dengan home
C.1.1	revisi stepping stone dari CRM	user	10	0	CRM delivered, revisi menu tidak signifikan
A.1.1	input numerik di inventori sering salah	end-user	30	30	ri-set input yang efektif
A.3.1	inventory menu tidak sesuai keinginan user	user	30	30	brainstorming kembali fitur inventory
A.1.1	inventory menu restruktur	dev	30	30	brainstorming kembali fitur inventory
SP0.6	inventory tidak delivered sprint 0	dev	30	30	pengajuan ekstensi sprint 0
SP0.7	sprint 0 extend 3 hari	operation	10	10	risiko rendah, karena user lebih mengutamakan website terlebih dahulu
		total	255	180	
		efektifitas	0,294118		

Dari pengamatan aktivitas manajemen risiko sprint 0 risiko residu dari note tasklist muncul sesuai ekspektasi, Sesuai ekspektasi Sprint 0 mengalami banyak insiden risiko karena tingkat maturitas produk masih sangat awal temuan efek dan risk point paling besar pada pembangunan modul inventory. tercatat dalam risk note yang diantisipasi, risiko baru bermunculan di masa sprint diberi kode SP dan terlihat 8 insiden dari 13 insiden diluar tasklist, sedangkan kode huruf risiko lainnya mengikuti kode note tasklist, total efektifitas penanganan risiko di sprint 0 adalah 0,294.

Pada Sprint 0 penanganan risiko didominasi 8 risiko ditransfer, risiko SP0.2, SP0.3 bisa ditangani diluar operasional, risiko A1.1 hingga SP0.7 ditangani dengan cara penambahan waktu, risiko lainnya ditangani dengan cara substitusi dan toleransi, seperti pada penanganan SP0.1,SP0.2, dan SP0.4 efektifitas penanganan risiko di fase sprint 0 adalah 0,294118

Di fase sprint 1 hasil rekapitulasi aktifitas manajemen risiko dirangkum pada tabel 5 berikut

TABEL 5
REKAPITULASI RISK NOTE EVALUASI SPRINT 1

kode tasklist	deskripsi	kategori	point risiko		mitigasi
			pra	pasca	
A.1.1	ada potensi supplier di take over pesaing	end-user	20	0	peta lokasi usaha dilampirkan, tapi tidak lokasi mitra
A.2.1	komoditi masih sedikit, prioritas web pada pemasaran bisnis	end-user	10	5	fitur ditangguhkan, katalog produk ditambahkan ketika komoditas >5 jenis
A.2.2	page terlalu penuh, mengurangi prinsip kemudahan akses pengguna	dev	10	5	fitur ditangguhkan, katalog produk ditambahkan ketika komoditas >5 jenis
A.2.3	demo multipage di publish di server odoo dikenakan charge fee lebih	devops	20	5	pembuatan website multipage ditangguhkan
B.2.1	data di fungsi lain ikut berubah mengikuti info rentang expected revenue	dev	30	0	revisi dipenuhi developer
B.2.2	perlu fitur akuisisi pengecekan otomatis nomor izin kemenperindag	dev	10	0	API service kemenperindag tidak tersedia, input info manual oleh user
C.1.1	mitra harus adaptasi skema input menggunakan messaging service telegram	end-user	30	10	mitra menyanggupi, pelatihan untuk minimasi kesalahan input teks
SP1.1	Sprint 1 modul inventori overtime 3 hari	operation	20	10	overtime dilakukan, delivery sprint 1 terlambat
		total	150	35	
		efektifitas	0,766667		

Dalam aktifitas sprint 1 risiko di sisi aktifitas *development* lebih sedikit daripada sprint 0 hal ini dikarenakan pada sprint 0 beberapa modul sudah mencapai status MVP, pada fase sprint 1 hanya revisi dan penambahan fitur, pembuatan modul inventory terkendala. sesuai tasklist C.1.1. dan SP.1.1, masih terdapat risiko dengan risk point

tinggi pada B.2.1 dan C.1.1 namun pada sprint 1 total risk point masih lebih rendah dari Sprint 0 awal pengembangan, dan terdapat 1 insiden diluar tasklist dengan kode SP. Total efektifitas penanganan risiko pun drastic naik disbanding sprint 0 yakni di angka 0,767.

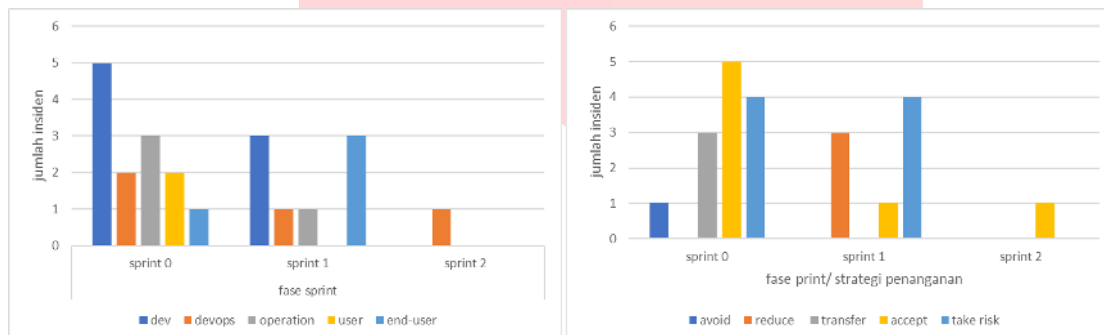
Aktifitas manajemen risiko Sprint 2 digambarkan pada tabel rekapitulasi pada Tabel 6

TABEL 6
REKAPITULASI RISK NOTE EVALUASI SPRINT 2

kode tasklist	deskripsi	kategori	point risiko		mitigasi
			pra	pasca	
D.1.1	deploy demo hanya bisa 1 akun	devops	20	0	user menyanggupi untuk sharing 1 akun untuk pemakaian bersama
		total	20	0	
		efektifitas	1		

Status dari masing masing modul yakni *website*, modul CRM dan modul Inventori dianggap telah MVP sehingga tim lebih fokus pada fase publish produk, pada fase ini produk dianggap stabil dan sudah memenuhi ekspektasi user sehingga tidak ditemukan risiko.

C. Pengukuran efektifitas manajemen risiko Datadibagian ini dirangkum dari hasil tabulasi Kanban risklog yang telah dirangkum di bagian 4.2. Pengamatan terhadap aktifitas penanganan risiko dilakukan sampai fase sprint 2 ketika produk sudah layak di publikasi. Total temuan dan penanganan risiko gambarkan pada grafik di gambar 17.

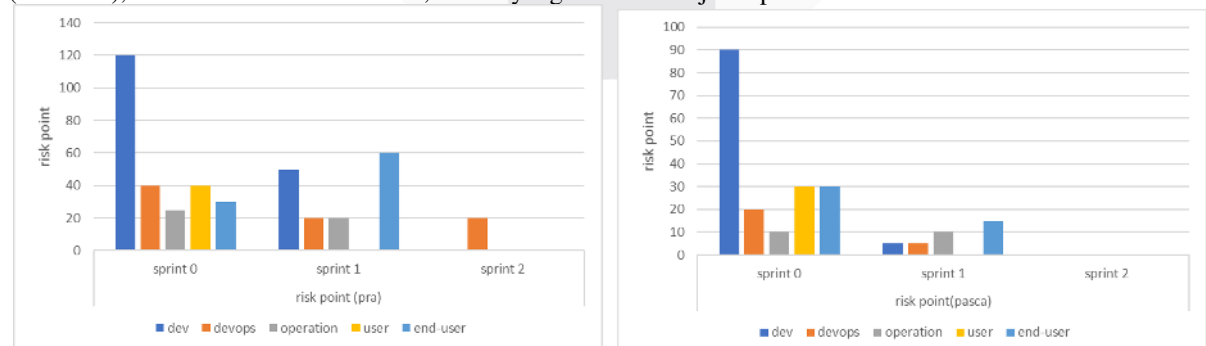


GAMBAR 17
GRAFIK TOTAL JUMLAH INSIDEN RISIKO TIAP SPRINT(KANAN) GRAFIK SEBARAN STRATEGI PENANGANAN RISIKO DITIAP SPRINT(KIRI)

Selama aktifitas penelitian dan pengamatan tercatat di fase sprint 0 terdapat 13 insiden risiko, di sprint 1 8 insiden dan di sprint 3 1 insiden, terdapat 8 insiden diluar tasklist di sprint 0 dan 1 insiden diluar tasklist di sprint 1, kelompok kategori insiden terdiri dari dev, devops, operation, user, dan end-user.

Menggunakan rencana penanganan risiko dijelaskan di bagian 3, pada sprint 0 lebih dominan risiko ditoleransi (accept) dan ditangani langsung (take risk), diikuti oleh transfer risiko, risiko yang

ditoleransi ditemukan dari sprint 0 didominasi oleh risiko pengembangan produk disaat *deployment*, demo atau publish. Risiko di kategori pengembangan produk (dev) ditangani langsung, namun untuk sisi operasional seperti keterlambatan atau perubahan fitur di sprint 0 lebih banyak ditransfer ke sprint 1 atau ke tim hosting/publish sedangkan insiden risiko. Lalu untuk ukuran keberhasilan strategi penanganan yang dilakukan disajikan pada Gambar 18

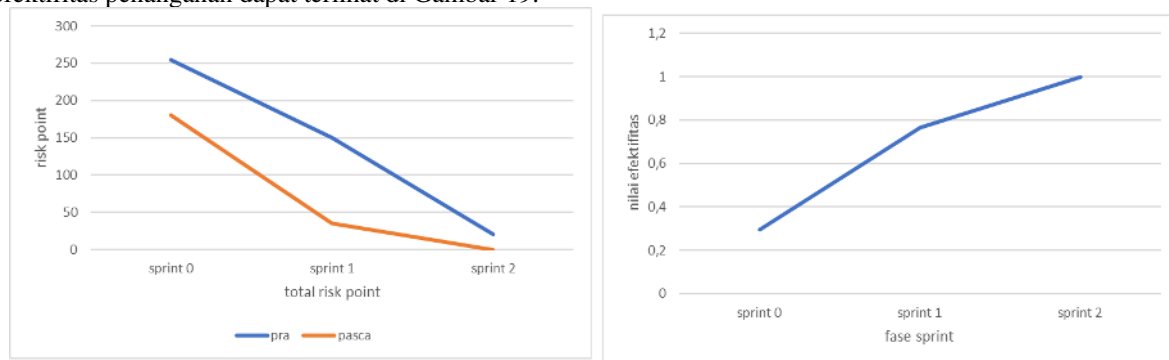


GAMBAR 18
GRAFIK TOTAL RISK POINT PRA MITIGASI (KIRI) DAN GRAFIK TOTAL RISK POINT PASCA MITIGASI (KANAN)

Tercatat kategori penyumbang risk point tertinggi pada dev Pada sprint 0 penanganan cenderung kurang efektif karena selisih risk point antara pra

mitigasi dan pasca mitigasi tidak berbeda jauh. Sedangkan organisasi dan tim sudah dapat stabil menangani insiden pada sprint 1 ditandai oleh

perbedaan selisih risiko yang jauh, gambaran efektifitas penanganan dapat terlihat di Gambar 19.



GAMBAR 19

GRAFIK PERBANDINGAN POIN PENANGANAN PRA MITIGASI DAN PASCA MITIGASI (KIRI) DAN GRAFIK PERBANDINGAN EFEKTIFITAS ANTARA FASE SPRINT (KANAN)

Selisih risk point pada gambar 19 grafik perbandingan poin penanganan pra mitigasi dan pasca mitigasi pada sprint 0 ialah 75 point dan pada sprint 1 115 point, terlihat dari 3 masa sprint risk point turun seiring dengan makin baiknya stabilitas produk dan koordinasi tim, begitu pula grafik perbandingan efektifitas antara fase sprint menunjukkan kenaikan.

V. KESIMPULAN

Metode risk poker berhasil mendokumentasi aktifitas risiko dengan dokumentasi hanya dengan Kanban risklog dari pengamatan lapangan pun tidak mengganggu kecepatan proses *Agile*, akan tetapi risk poker tidak dapat mencatat potensi risiko yang berasal diluar dari tasklist atau sprint *backlog*. Sehingga terjadi 8 insiden di sprint 0 dan 1 insiden tidak terduga di sprint 1 yang tidak diduga oleh tim. efektifitas penanganan berturut turut dari sprint 0, sprint 1 hingga sprint 2 adalah 0.2941, 0.7667 dan 1, penanganan insiden cukup cepat namun efektifitas di sprint 0 rendah karena banyak insiden yang berasal dari luar tasklist.

Dari pengamatan lapangan dokumentasi risk notes memakan waktu dan tenaga SCRUM master akan tetapi pembuatan risk notes secara langsung mempengaruhi pembentukan sprint backlog pada tiap fase menjadi lebih *feasible* dilakukan meminimasi revisi di sprint selanjutnya, mengurangi cost pengembangan dan mencapai masa maturisasi lebih cepat dari dugaan user.

REFERENSI

- [1] Kulkarni, R. H., Padmanabham, P., Harshe, M., Baseer, K. K., & Patil, P. 2017 . Investigating Agile Adaptation for Project Development. 7(3), IJECE
- [2] Hammad, M., Inayat, I., & Zahid, M. 2019. Risk management in agile software development: A survey. International Conference on Frontiers of Information Technology

- [3] Lopes. S. S., Grat,R,C., Contessoto, A., Oliveira. A.L., Braga.R.T.V.. 2021. A Risk Management Framework for Scrum Projects, 23rd International Conference on Enterprise Information Systems

- [4] Mamoghli. S., Cassivi. L.. 2019. Agile ERP Implementation: The Case of a SME, ICEIS

- [5] Inayat. M. H. I.2018. Integrating Risk Management in Scrum Framework (FIT)

- [6] Rygge. H.,Josang. A..2018. Threat Poker: Solving Security and Privacy Threats in Agile Software Development, NordSec

- [7]Software Development Risk Management Model- a goal-driven approach Shareeful Islam Institut fur Informatik Technische Universitat Munchen 2011

- [8] Albadarneh.A.,Albadarneh.I., Qusef.A.. 2015.] Risk Management in Agile Software Development: a Comparative Study, IEEE/ Jordan Conference AEECT

- [9] Ghazali.S.N.H.,Salim.S.S.,Inayat.I.,Hamid.S.H. A..2018. A Risk Poker Based Testing Model For Scrum , international journal of computer

- [10] Moran.A..2016. Risk Management in Agile Projects.ISACA vol 2

- [11] Wanderleya.M., Menezes.J., Gusmao.C.,Limaa.F..2015.Proposal of risk management metrics for multiple project software development,CENTERIS / ProjMAN / HCist

- [12] odoo.com diakses pada 12.00 WIB 09 september 2022

- [13] <https://digital.ai/glossary/agile-development-success> diakses pada 12.00 18 september 2022

- [14] Buganová.K., Šimíčková.J..2019. Risk management in traditional and agile project management TRANSCOM
- [15] Shankarmani.R., Pawar.R., Mantha.S.S..2012. Agile Methodology Adoption: Benefits and Constraints. International Journal of Computer Applications Volume 58– No.15
- [16] Schön.E-M., Radtke.D.,Jordan.C..2020. Improving Risk Management in a Scaled Agile Environment. International Conference on Agile Software Development
- [17] Tavaresa.B.G., da Silva.C.E.S., de Souza.A.D..2017. Risk management analysis in Scrum software projects. 2017 International Federation of Operational Research Societies
- [18] <https://www.softwaretestinghelp.com/planning-poker-scrum-poker-cards-agile-estimation/> diakses pada 12.00 18 september 2022
- [19] Coyle.S.,Conboy.k..2015. A Case Study of Risk Management in Agile Systems Development. Researchgate.net
- [20] <https://medium.com/agileinsider/agile-vs-waterfall-in-a-nutshell-41cf76fd4763> diakses pada 12.00 18 september 2022
- [21] Future of risk management in financial services: Integrating Risk management and Agile Projects. Deloitte. diakses pada 18 september 2022
- [22] <https://www.vivifyscrum.com/insights/risk-management-scrum> diakses pada 18 september 2022
- [23] Fagarasan.C.,Popa.O.,Pisla.A.,Cristea.C.. 2021..Agile, waterfall and iterative approach in information technology projects. IOP
- [24]<https://content.intland.com/blog/sdlc/risk-management-in-agile-and-waterfall-environments> diakses pada 25 september 2022
- [25]<https://www.edrawsoft.com/agile-vs-waterfall.html> diakses pada 25 september 2022
- [26]Andrei.B.A.,Casu-pop.A.C.,Gheorghe.S.C., Boiangiu.C.A..2019. A STUDY ON USING WATERFALL AND AGILE METHODS IN SOFTWARE PROJECT MANAGEMENT. Polytechnic University of Bucarest
- [27]Jabeen.R., Awan.M.D..2016. Role of Risk Management in Scrum. Communications on Applied Electronics(CAE)
- [28]<https://www.lumeer.io/waterfall-project-management/> diakses pada 25 september 2022
- [29]<https://hygger.io/blog/the-risks-of-waterfall-methodology/> diakses pada 25 september 2022