

BAB II KAJIAN LITERATUR

2.1 Landasan Teori

2.1.1 Chilli Indonesia

Chilli Indonesia merupakan *startup* yang khusus menangani permasalahan cabai di Indonesia. Chilli Indonesia merancang sistem bisnis pertanian terintegrasi dari hulu hingga hilir khusus komoditas cabai. Dimulai dari sisi *on farm, off farm, food processing* hingga IT [4].

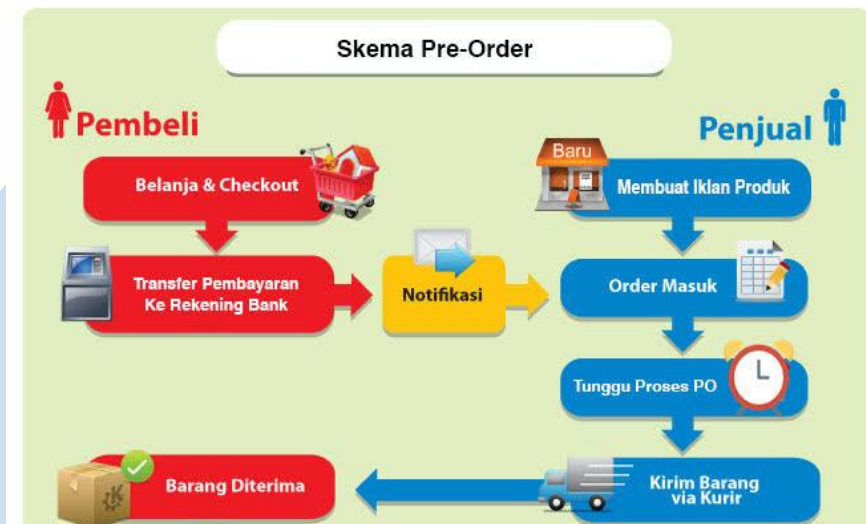
Pada sisi hulu, Chilli Indonesia menerapkan Standar Operasional Prosedur (SOP) dan *Good Agricultural Practices* (GAP), baik pada sistem pertanian konvensional dan teknologi *Greenhouse*. Selain itu, untuk memenuhi kebutuhan pasar maka kami menerapkan kalender tanam serta pola tanam sehingga lonjakan kebutuhan cabai bisa terjaga [4].

Pada sisi hilir, Chilli Indonesia memproduksi produk olahan dari cabai. Produk olahan untuk kebutuhan konsumsi hingga obat dan kosmetik. Produk yang akan difokuskan pada tahap awal adalah produksi sambal beserta produk turunan cabai untuk konsumsi dengan merk Rumah Sambal yang diproduksi sesuai prinsip *Good Manufacturing Practice* (GMP) [4].

Sedangkan pada sisi IT, menjadi solusi keseluruhan dari hulu hingga hilir untuk mengatasi masalah cabai. *Platform* ini bisa digunakan bagi petani, UMKM, pembeli, serta investor. *Platform* dapat digunakan sebagai *marketplace*, kemudian petani atau penggiat UMKM juga dapat mendapatkan modal usaha dari investor dengan akad salam serta mendapatkan materi-materi pengetahuan dan praktis yang berkaitan dengan cabai pada *platform* ini [4].

2.1.2 Pre-Order

Pre-Order (PO) adalah sistem pembelian barang dengan memesan dan membayar terlebih dahulu sebelum produksi dimulai, dengan tenggang waktu tunggu (estimasi/perkiraan) sampai barang tersedia [6].



Gambar 2.1 Skema *Pre-Order* pada *market place*

2.1.3 Akad Salam

Salam berasal dari kata *As-salaf* yang artinya pendahuluan karena pemesan barang menyerahkan uangnya di muka. Para *fuqaha* menamainya *al-mahawi'ij* (barang-barang mendesak) karena ia sejenis jual-beli yang dilakukan mendesak walaupun barang yang diperjualbelikan tidak ada di tempat. "Mendesak", dilihat dari sisi pembeli karena ia sangat membutuhkan barang tersebut di kemudian hari sementara dari sisi penjual, ia sangat membutuhkan [7].

Salam dapat didefinisikan sebagai transaksi atau akad jual beli barang yang diperjualbelikan belum ada ketika transaksi dilakukan, dan pembeli melakukan pembayaran di muka sedangkan penyerahan barang baru dilakukan di kemudian hari. PSAK 103 mendefinisikan salam sebagai akad jual beli barang pesanan (*muslim fiih*) dengan pengiriman dikemudian hari oleh penjual (*muslam illaihi*) dan pelunasannya dilakukan oleh pembeli (*al-muslim*) pada saat akad disepakati sesuai dengan syarat-syarat tertentu [7]. Syarat – syaratnya ialah sebagai berikut :

1. Ketentuan tentang Pembayaran:

- a. Alat bayar harus diketahui jumlah dan bentuknya, baik berupa uang, barang, atau manfaat.
- b. Pembayaran harus dilakukan pada saat kontrak disepakati.
- c. Pembayaran tidak boleh dalam bentuk pembebasan hutang.

2. Ketentuan tentang Barang:

- a. Harus jelas ciri-cirinya dan dapat diakui sebagai hutang.
- b. Harus dapat dijelaskan spesifikasinya.
- c. Penyerahannya dilakukan kemudian.
- d. Waktu dan tempat penyerahan barang harus ditetapkan berdasarkan kesepakatan.
- e. Pembeli tidak boleh menjual barang sebelum menerimanya.
- f. Tidak boleh menukar barang, kecuali dengan barang sejenis sesuai kesepakatan.

3. Ketentuan tentang Salam Paralel dipebolehkan melakukan salam paralel dengan syarat, akad kedua terpisah dari, dan tidak berkaitan dengan akad pertama.

4. Penyerahan Barang Sebelum atau pada waktunya:

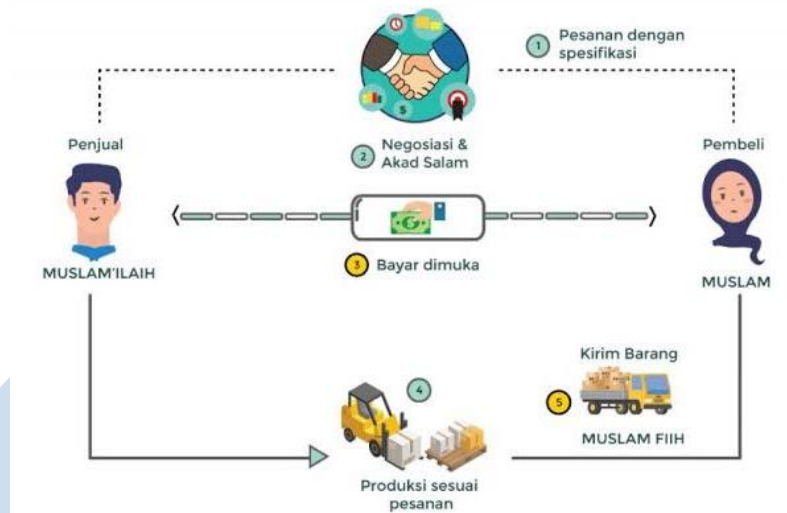
- a. Penjual harus menyerahkan barang tepat pada waktunya dengan kualitas dan jumlah yang telah disepakati.
- b. Jika penjual menyerahkan barang dengan kualitas yang lebih tinggi, penjual tidak boleh meminta tambahan harga.
- c. Jika penjual menyerahkan barang dengan kualitas yang lebih rendah, dan pembeli rela menerimanya, maka ia tidak boleh menuntut pengurangan harga (diskon).
- d. Penjual dapat menyerahkan barang lebih cepat dari waktu yang disepakati dengan syarat kualitas dan jumlah barang sesuai dengan kesepakatan, dan ia tidak boleh menuntut tambahan harga.

- e. Jika semua atau sebagian barang tidak tersedia pada waktu penyerahan, atau kualitasnya lebih rendah dan pembeli tidak rela menerimanya, maka ia memiliki dua pilihan:
 - i. membatalkan kontrak dan meminta kembali uangnya,
 - ii. menunggu sampai barang tersedia.
5. Pembatalan Kontrak, pada dasarnya pembatalan *salam* boleh dilakukan, selama tidak merugikan kedua belah pihak.
6. Perselisihan, jika terjadi perselisihan di antara kedua belah pihak, maka persoalannya diselesaikan melalui Badan Arbitrasi Syari'ah setelah tidak tercapai kesepakatan melalui musyawarah [8].

Dalam *murabahah*, kita kenal ada penjualan tangguh yang artinya barang diserahkan terlebih dahulu sedangkan pembayaran kemudian. Salam merupakan kebalikannya, pembayaran dilakukan terlebih dahulu dan penyerahan barang dilakukan kemudian. Untuk menghindari risiko yang merugikan, pembeli boleh meminta jaminan dari penjual [7]. Dalam hadis riwayat Bukhari dari Ibn 'Abbas, Nabi bersabda:

وَوَزْنٍ مَّعْلُومٍ مِّنْ أَسْلَفٍ فِي شَيْءٍ فَفِي كَيْلٍ مَّعْلُومٍ
إِلَى أَجَلٍ مَّعْلُومٍ.

“Barang siapa melakukan *salaf* (salam), hendaknya ia melakukan dengan takaran yang jelas dan timbangan yang jelas, untuk jangka waktu yang diketahui.” (HR. Bukhari, Shahih al-Bukhari [Beirut: Dar al-Fikr, 1955], jilid 2, h. 36) [8].



Gambar 2.2 Skema Akad Salam

Manfaat transaksi salam bagi pembeli adalah adanya jaminan memperoleh barang dalam jumlah dan kualitas tertentu pada saat ia membutuhkan dengan harga yang disepakatinya diawal. Sementara manfaat bagi penjual adalah diperolehnya dana untuk melakukan aktivitas produksi dan memenuhi sebagian kebutuhan hidupnya [7].

2.1.4 *Unified Modeling Language (UML)*

UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis obyek. UML merupakan singkatan dari *Unified Modeling Language*. (UML) juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena *developer* harus melakukan penelusuran dan mempelajari kode program. UML juga dapat menjadi alat bantu untuk *transfer* ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainnya. Tidak hanya antar *developer* terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya UML [9].

Pada UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. UML mendefinisikan diagram-diagram sebagai berikut:

2.1.4.1 Usecase Diagram

Usecase diagram menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. yang menjadi persoalan itu apa yang dilakukan bukan bagaimana melakukannya. *Usecase diagram* dekat kaitannya dengan kejadian-kejadian. Kejadian (*scenario*) merupakan contoh apa yang terjadi ketika seseorang berinteraksi dengan sistem [10].

2.1.4.2 Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan obyek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain [10].

2.1.4.3 Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses *paralel* yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, yang sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*).

Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari *level* atas

secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas [10].

2.1.5 User Story

User story merupakan salah satu hal utama yang ada pada *Arifacts Scrum*. *User story* adalah kebutuhan yang paling tinggi dan berisi informasi-informasi yang dapat membantu *developer* dalam menjalankan dan mengestimasi waktu pada produk atau proyek yang sedang berlangsung [11].

2.1.6 Pengembangan Model Scrum

Scrum dibangun di atas teori proses kontrol empiris atau bisa disebut empirisme. Empirisme menyatakan bahwa pengetahuan datang dari pengalaman dan pengambilan keputusan didasari oleh apa yang telah diketahui hingga saat ini. Scrum menggunakan pendekatan yang bertahap dan berkelanjutan untuk mengoptimalkan kemampuan prediksi dan mengendalikan risiko. Tiga pilar yang memperkokoh setiap implementasi dari proses kontrol empiris adalah transparansi, inspeksi, dan adaptasi [12].

a) Transparansi

Aspek signifikan dari sebuah proses harus dapat dilihat oleh orang-orang yang bertanggung jawab terhadap dampaknya. Transparansi membutuhkan aspek-aspek tersebut ditentukan oleh standar baku sehingga para pengamat memiliki pemahaman yang sama terhadap apa yang sedang ditinjau.

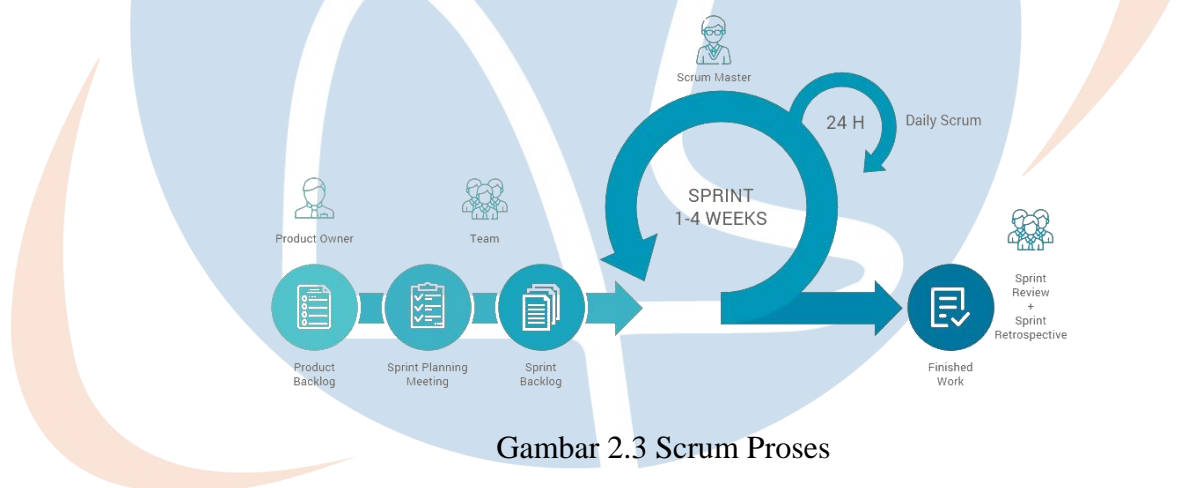
b) Inspeksi

Pengguna Scrum harus sering menginspeksi *Arifacts Scrum* dan perkembangan menuju *Sprint Goal* agar mereka dapat mendeteksi adanya variansi hasil yang tidak diharapkan. Proses inspeksi juga disarankan tidak dilakukan terlalu sering sampai menghambat pekerjaan. Inspeksi akan sangat bermanfaat jika dilakukan oleh pemeriksa yang kompeten disaat pekerjaan tersebut sedang berada.

c) Adaptasi

Jika pemeriksa menemukan bahwa ada satu hal atau lebih dari proses yang menyimpang di luar ambang batas yang bisa diterima yang dapat menyebabkan produk tidak bisa diterima, maka proses atau materi yang sedang diproses harus diubah. Perubahan harus dilakukan secepatnya untuk meminimalkan penyimpangan yang semakin jauh. Scrum memiliki empat acara formal untuk melakukan inspeksi dan adaptasi, seperti yang dijabarkan di dalam dokumen ini yakni:

- *Sprint Planning*
- *Daily Scrum*
- *Sprint Review*
- *Sprint Retrospective*



Gambar 2.3 Scrum Proses

2.1.6.1 Komponen Scrum

Scrum memiliki tiga peranan penting yaitu *Product Owner*, *Development Team*, dan *Scrum master* yaitu:

- a) *Product Owner* adalah orang yang bertanggung jawab untuk memaksimalkan nilai bisnis dari produk yang dihasilkan oleh *Development Team*. Cara melakukannya sangat bervariasi antar organisasi, *Scrum Team*, dan individu [12].
- b) *Development Team* terdiri dari para ahli profesi yang bekerja untuk menghantarkan Increment “Selesai” yang berpotensi untuk dirilis

disetiap akhir *Sprint Increment* “Selesai” wajib tersedia pada saat *Sprint Review*. Hanya anggota dari *Development Team* yang membuat *Increment* ini.

Development Team dibentuk dan diberikan wewenang oleh organisasi untuk menyusun dan mengelola pekerjaan mereka sendiri. Hasil sinergi dari tim ini akan mengoptimalkan efisiensi dan efektivitas *Development Team* secara keseluruhan [12].

- c) *Scrum Master* bertanggung jawab untuk mengenalkan dan menyokong penggunaan Scrum sebagaimana dijelaskan di dalam Panduan Scrum ini. *Scrum Master* melakukan ini dengan membantu orang-orang agar dapat memahami teori, praktik-praktik, aturan-aturan dan tata nilai Scrum.

Scrum Master adalah pemimpin yang melayani *Scrum Team*. *Scrum Master* membantu orang-orang di luar *Scrum Team* untuk dapat memahami interaksi mana yang bermanfaat dan tidak bermanfaat. *Scrum Master* membantu orang-orang untuk mengubah interaksi ini guna memaksimalkan nilai bisnis yang dihasilkan oleh *Scrum Team* [12].

2.1.7 ReactJs

ReactJs adalah sebuah pustaka/*library* Javascript yang bersifat *opensource* untuk membangun *user interface* yang dibuat oleh Facebook. ReactJs hanya mengurus semua hal yang berkaitan dengan tampilan dan logika di sekitarnya. ReactJs ini diciptakan dengan tujuan untuk membangun aplikasi skala besar dengan data yang berubah dan terus berubah dari waktu ke waktu [13]. Berikut merupakan beberapa keunggulan ReactJs:

a) *Declarative*

ReactJs membuatnya proses pembuatan *user interface* menjadi interaktif. Rancang tampilan sederhana untuk setiap *state* di aplikasi, dan ReactJs secara efisien akan memperbarui dan membuat komponen yang tepat ketika data mengalami perubahan.

Tampilan deklaratif menjalankan kode menjadi lebih mudah diprediksi dan lebih mudah untuk di-*debug* [14].

b) *Component-Based*

Build encapsulated components yang mengelola status *state*-nya sendiri, kemudian saat pengembangnya menggabungkan komponen-komponen tersebut untuk membentuk *user interface* yang kompleks. Selain itu, karena logika pada komponen ReactJs ditulis langsung menggunakan Javascript (bukan menggunakan template), maka dapat dengan mudah mengoper data melalui aplikasi dan menempatkan *state* agar tetap berada di luar DOM [14].

c) ReactJs juga dapat digunakan untuk *me-render* di *server* menggunakan Node [13].

d) ReactJs memungkinkan untuk dapat berinteraksi dengan *library* dan *framework* lain [13].

2.1.8 NodeJs

NodeJs adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis *website* dan ditulis dalam sintaks bahasa pemrograman Javascript. Bila selama ini kita mengenal Javascript sebagai bahasa pemrograman yang berjalan di sisi *client/browser* saja, maka NodeJs ada untuk melengkapi peran Javascript sehingga bisa juga berlaku sebagai bahasa pemrograman yang berjalan di sisi *server*, seperti halnya PHP, Ruby, Perl, dan sebagainya. NodeJs memiliki pustaka *server* HTTP sendiri sehingga memungkinkan untuk menjalankan *server web* tanpa menggunakan program *server web* seperti *Apache* atau *Nginx* [15].

Untuk mengeksekusi Javascript sebagai bahasa *server* diperlukan *engine* yang cepat dan mempunyai performansi yang bagus. *Engine* Javascript dari Google bernama V8-lah yang dipakai oleh NodeJs yang juga merupakan *engine* yang dipakai oleh *browser* Google Chrome [16].

Aplikasi NodeJs dijalankan dalam satu proses tunggal, tanpa membuat *thread* yang baru untuk setiap *request*. NodeJs menyediakan satu set perangkat I / O asinkron yang primitif dalam pustaka standarnya yang akan mencegah kode Javascript dari pemblokiran yang sering terjadi pada bahasa pemrograman sisi *server* pada umumnya, pustaka di NodeJs ditulis dengan menggunakan paradigma *non-blocking* yang menjadikan perilaku pemblokiran sebagai pengecualian [16].

Ketika NodeJs perlu melakukan operasi I / O, seperti kegiatan membaca dari jaringan, mengakses database atau sistem *file*, daripada memblokir *thread* dan memperlama proses siklus menunggu dari CPU, NodeJs akan melanjutkan operasi ketika respon sudah diterima [16].

Hal ini memungkinkan NodeJs untuk menangani ribuan koneksi secara bersamaan dengan satu *server* tanpa menghasilkan beban dalam mengelola konkurensi *thread*, yang bisa menjadi sumber *bug* yang signifikan [16].

NodeJs memiliki keunggulan yang unik karena jutaan pengembang frontend yang menulis Javascript untuk *browser* sekarang dapat menulis kode pada sisi *server* selain kode pada sisi klien tanpa perlu mempelajari bahasa yang berbeda [16].

2.2 Metode Pengujian

2.2.1 Blackbox Testing

Blackbox Testing berfokus pada pengujian spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi *input* dan *output*, serta melakukan pengetesan pada spesifikasi fungsional program.

2.2.2 Metode Skala Pengukuran *Likert*

Skala Likert digunakan untuk mengukur sikap atau pendapat seseorang atau sejumlah kelompok terhadap sebuah fenomena sosial yang jawaban setiap item instrumen mempunyai gradasi dari sangat positif sampai sangat negatif. Dengan *Skala Likert* variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator tersebut dijadikan titik tolak untuk menyusun item-item instrumen yang dapat berupa pernyataan atau pertanyaan. Skala yang dipakai pada penelitian ini mencakup empat unsur penilaian; Tidak Setuju, Kurang Setuju, Setuju, Sangat Setuju.

2.2.3 *User Acceptance Testing*

User Acceptance testing (UAT) adalah proses untuk mendapatkan konfirmasi bahwa sebuah sistem memenuhi yang disepakati persyaratan dan mengetes apakah semua fungsi dan fitur berjalan dengan baik atau tidak. Caranya, *user* mencoba *software* untuk sistem pendukung keputusan terhadap semua kondisi data dan mencocokkannya dengan hasil yang diharapkan. Apabila hasil semua tes sesuai dengan keluaran yang diharapkan, maka tes tersebut dinyatakan berhasil. Apabila ada beberapa fitur yang tidak memberikan keluaran yang diharapkan, maka aplikasi diperbaiki untuk disesuaikan dengan keluaran yang diharapkan atau ditolak.

STT - NF

2.3 Penelitian Terkait

Dalam penyusunan proposal tugas akhir ini, dilakukan penelitian terhadap fitur *pre-order* dengan metode akad salam yang sejenis untuk transaksi jual-beli online, berikut beberapa penelitian terkait yang ditemukan:

Tabel 2.1 Penelitian Terkait

Nama Peneliti	Judul Penelitian	Fitur	Metodologi Penelitian	Hasil Penelitian
		Pembayaran		
Rohman, Muhammad Amru	ANALISIS APLIKASI AKAD AS-SALAM DALAM SISTEM JUAL BELI ONLINE DI SUPPLIER HERBAL MURAH SURABAYA	Menggunakan metode pembayaran biasa (pembeli transfer uang ke nomer rekening yang dicantumkan penjual pada halaman penjualan)	Metode studi kasus deskriptif - analisis	Hasil penelitian menyimpulkan bahwa hubungan para pihak di dalam perjanjian akad salam secara online (melalui <i>market place</i>) sama saja dengan perjanjian akad salam seperti biasanya. Namun, akad salam dalam <i>market place</i> tidak ada temu muka diantara pembeli dan penjual, hanya saja pelaku akad dipertemukan dalam satu situs jaringan internet.

Ainiyah, Qurrotul	ANALISIS HUKUM ISLAM TERHADAP JUAL BELI DENGAN CARA <i>PRE-ORDER</i> DI TOKO <i>ONLINE TANJUNG SPORT</i>	Menggunakan metode pembayaran biasa (pembeli transfer uang ke nomer rekening yang dicantumkan penjual pada halaman penjualan)	Metode studi kasus analisis - deskriptif	Jual beli dengan menggunakan sistem <i>online</i> yang dipraktikkan oleh toko <i>online Comfortable Clothing</i> Sidoarjo adalah menggunakan salah satu akad jual beli yaitu Salam atau dengan cara <i>pre-order</i> . penjual dan pembeli melakukan transaksi secara tidak langsung (tidak dalam satu majlis) yaitu melalui media internet baik <i>website</i> maupun instagram
Laila, Eka Nikmatul	IMPLEMENTASI SISTEM <i>PRE-ORDER</i> PADA <i>ONLINE SHOP VIE_DIE PRODUCTION</i> DI KECAMATAN SEMEN KABUPATEN KEDIRI PERSPEKTIF EKONOMI SYARI'AH	Menggunakan metode pembayaran biasa (pembeli transfer uang ke nomer rekening yang dicantumkan penjual pada	Metode studi kasus kualitatif	Hasil penelitian menyimpulkan bahwa penggunaan <i>pre-order</i> – ebelum terjadinya proses produksi dengan menyebutkan spesifikasi barang. Apabila terjadi kerusakan pada barang, pembelian ditanggung oleh penjual. Praktik pada online shop <i>VIE_DIE Production</i> yang

		halaman penjualan)		sudah sesuai dengan konsep salam dalam ekonomi syariah adalah barang selalu dikirim dan tidak ada unsur penipuan akan tetapi terkadang melebihi batas waktu akad, menyebutkan spesifikasi barang, kesepakatan antara penjual dan pembeli.
Hilmi, Muhammad Fadhil	PENGEMBANGAN FITUR <i>PRE-ORDER</i> MENGUNAKAN METODE AKAD SALAM SEBAGAI MEDIA TRANSAKSI JUAL- BELI PRODUK CABAI STUDI KASUS : CHILI.ID	<i>Payment gateway</i>	Metode studi kasus kualitatif - kuantitatif	

STT - NF