

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 TINJAUAN PUSTAKA

Pada penulisan penelitian ini peneliti menggali informasi dari skripsi dan literasi ilmiah yang sudah ada sebelumnya. Ini bertujuan untuk dijadikan perbandingan baik kelebihan dan kekurangan penelitian sebelumnya. Peneliti juga mencari informasi melalui buku-buku maupun skripsi untuk mencari informasi mengenai teori yang berkaitan dengan judul yang dijadikan untuk landasan teori. Adapun karya ilmiah yang penulis maksud adalah sebagai berikut:

Syifa Nur Rakhmah, Rizki Aulianita, Ganda Wijaya, Minda Septiani (2023), melakukan penelitian dengan judul rancang bangun toko gadget berbasis *website* menggunakan metode *scrum*. Pada penelitian ini berfokus pada masalah toko gadget yang tidak melakukan pemasaran online dan hanya mengandalkan penjualan dari toko offline. Dengan dibuatkan *website* untuk toko gadget ini diharapkan bisa menaikkan omset penjualan dan pencatatan penjualan yang lebih rapi. Metode yang digunakan yaitu metode *SCRUM*, metode ini dipilih karena bisa menciptakan *software* dengan kualitas yang baik. Bisa dilihat dari kesimpulan metode *SCRUM* bisa membuat website lebih efisien dari segi waktu dan biaya. Selain itu, *SCRUM* juga banyak digunakan di industri yang besar. Karena kerangka kerja *SCRUM* mudah diterapkan dan bisa dibagi-bagi dalam tim jadi cukup adaptif dengan perubahan. Dengan adanya report dari tiap task yang dikerjakan, jadi mempermudah dalam memantau jalanya perancangan sistem[8].

Audyna Renata, Marsyanda Razita Zahrani, Muhamad Bintang Melandri (2022) melakukan penelitian dengan judul analisis efisiensi penggunaan manajemen proyek STI (*Agile, Scrum dan Waterfall*) pada suatu Perusahaan atau organisasi. Pada penelitian ini mencoba membandingkan kerangka kerja *scrum* dan *waterfall* dalam proses pengembangan *software*. Dijelaskan bahwa kerangka kerja *scrum* yang lebih *fleksibel* dan selain sebagai model pengembangan

software juga bisa digunakan sebagai manajemen pengembangan. Dibanding dengan *waterfall* ini memiliki kerangka kerja yang bertolak belakang. Dengan sifat *waterfall* yang *statis* dengan entitas yang beraturan. Dalam proses perancangannya menerapkan alur kerja secara berurutan dan *sekuensial*. Dari tahap pengujian yang telah dilakukan dalam penelitian tersebut, diambil Kesimpulan bahwa *metodelogi scrum* dalam manajemen proyek lebih efisien dibanding metode *waterfall*[9].

V A Sugianto, W K Mahardika, Y Wijayanti (2023) melakukan penelitian dengan judul strategi pemasaran digital pada ayam panggan sunmor menggunakan model *sostac*. Pada penelitian tersebut membuktikan bahwa *digital marketing* menggunakan *metode sostac model* bisa diterapkan untuk menganalisis dan menentukan strategi bisnis *UMKM*. Hasil penelitian yang dilakukan dengan menggunakan metode *sostac model* bisa dimanfaatkan untuk menentukan perencanaan pemasaran yang terarah guna untuk mengembangkan bisnis. Hasil dari penelitian tersebut bahwa *UMKM* ayam bakar sunmor bisa mengoptimalkan media sosial sebagai media untuk pemasaran. Selain itu memperhatikan lokasi dengan mempertimbangkan kapasitas pelanggan yang *dine in* dan tempat parkir sehingga tingkat penjualan akan lebih optimal. Fokus penelitian ini lebih ke memaksimalkan pemasaran dan penjualan, penerapan *sostac model* tepat karena bisa mencapai tujuan tersebut[10]

Vania Valentina Pandora dan Yenni Merlin Djajalaksana (2022), Penelitian yang dilakukan mengenai penerapan *digital marketing multichannel* dalam pemasaran program studi sistem informasi. Penerapan ini dilakukan melalui Instagram, facebook, website dengan *landing page* dan *email marketing*. Untuk menghasilkan sebuah data yang akurat dan bisa menyesuaikan dengan target, peneliti mengevaluasi hasil penelitiannya menggunakan Google Analytics, Instagram Insight, Facebook Insight, Facebook Pixel, dan Email Campaign Report [1].

Ayu Rizkita Agung, Saifudin Husen, Aries Saifudin (2024) melakukan penelitian dengan judul pengujian sistem informasi *inventaris* pada Perusahaan *distributor* farmasi menggunakan metode *blackbox testing*. Penelitian tersebut

mengungkapkan bahwa pengujian blackbox testing hanya berfokus pada pengujian website secara fungsionalitasnya saja, tanpa memperhatikan implementasi kode program. Selain itu input dan output website menjadi poin penting dalam proses pengujian. Karena kita harus memastikan apakah input yang kita perintahkan apakah outputnya sesuai dengan yang diharapkan. Hasil penelitian ini juga mengatakan bahwa metode blackbox testing dianggap efektif dalam mengidentifikasi potensi kesalahan atau kelemahan dalam sistem. Peneliti menyimpulkan *blackbox testing* memungkinkan pengujian sistem inventaris dari perspektif eksternal. Dengan cara tersebut kelemahan dalam antarmuka pengguna dan *fungsionalitas* yang tidak sesuai dapat terdeteksi sebelum peluncuran sistem[11].

Berdasarkan beberapa tinjauan Pustaka diatas peneliti akan membangun sebuah *website* yang akan digunakan sebagai media pemasaran digital. Nantinya pemasaran digital akan dijalankan melalui instagram ads dan facebook ads yang dikelola lewat meta business suite. Website digunakan sebagai media bagi audiens untuk mencari informasi detail produk dan melakukan pemesanan. *Website* akan dibangun dengan sederhana dan cenderung memudahkan calon customer untuk melakukan pembelian. *Digital marketing* berperan penting dalam menyasar target *audiens* yang relevan dengan produk yang ditawarkan. Penggunaan pemasaran digital *multichannel* akan diterapkan untuk mengumpulkan data, sehingga kita memperoleh data target konsumen yang terukur dan akurat.

Pemakaian framework *SOSTAC model* digunakan dalam proses perancangan selama kampanye. Beberapa tahapan yang ada pada *SOSTAC model* dapat dijadikan tolak ukur, apakah target yang sudah ditetapkan bisa tercapai selama kampanye. Selain itu untuk mengetahui apakah sistem website yang dibuat bisa dijalankan dengan baik atau tidak, dilakukan dengan pengujian sistem. Pengujian sistem yang dipilih adalah metode *blackbox testing*, karena bisa menguji sistem secara fungsionalitas. Metode *blackbox testing* juga cenderung lebih mudah dikerjakan karena hanya menguji fungsionalitas sistem tanpa melihat alur kode programnya. Pengujian sistem dilakukan guna

memastikan bahwa sistem yang sudah dibangun bisa dijalankan dengan baik sebelum sistem diluncurkan.

2.2 LANDASAN TEORI

2.2.1 Sistem Informasi

Sistem merupakan himpunan suatu unsur, *variable* yang terorganisir, saling berinteraksi dan bergantung satu sama lain. Informasi adalah data yang telah diolah serta dimanfaatkan oleh pengguna sebagai alat pengambil keputusan dan mendukung sumber informasi. Sistem informasi merupakan kombinasi antara manusia, *hardware*, *software*, *jaringan komunikasi* dan sumber daya data yang dikumpulkan dan disebar dalam sebuah organisasi. *Website* merupakan *server side scripting* yang berisi perintah program yang berjalan disisi *web server* yang kemudian dikirim ke *browser* sebagai *client side scripting*. Website memiliki sebuah url dimana dapat digunakan untuk membuka halaman satu ke halaman lainnya untuk mencari informasi [12]. Banyak komponen membangun *website* agar dapat digunakan, mulai dari bahasa *pemrograman*, *database* dll.

- a. PHP (*Hypertext Proccesor*) merupakan sebuah bahasa pemrograman berupa script yang diperlukan dalam pembuatan halaman *website dinamis*. Artinya halaman website yang dibuat atas permintaan client. *PHP (server side HTML embedded scripting)* adalah script yang terintegrasi dengan HTML dan berjalan di server[13].
- b. Bootstrap merupakan *framework css* yang dibuat oleh Mark Okto dan Jacob Thronton. *Bootsrap* ini bisa dikatakan template *website* berisi *framework HTML, CSS dan Javascript* yang digunakan untuk membuat *website* yang responsif. *Bootstrap* juga mudah untuk digunakan, cukup tau dasar penggunaan *HTML* dan *CSS*. Karena *bootstrap* diciptakan untuk mempermudah dalam mendesain web mulai dari level pemula sampai yang sudah mempunyai pengalaman [14].
- c. *Javascript* adalah bahasa pemrograman yang digunakan untuk membuat fitur-fitur interaktif, seperti melakukan perubahan pada konten *website* yang tidak bisa di cover oleh *HTML* dan *CSS*. Pada implementasinya

javascript dikolaborasikan dengan *HMTL* dan *CCS* dalam merancang sebuah *website*[15].

- d. Basis Data terdiri dari 2 kata basis dan data, basis artinya penyimpanan sedangkan data adalah gambaran dari sebuah fakta yang mewakili objek. *Basis Data* merupakan sebuah himpunan kelompok data yang diorganisasikan dan di simpan sedemikian rupa untuk mencukupi berbagai kebutuhan dengan mudah[16].
- e. MySQL (*My Structured Query Language*) merupakan sebuah basis data RDBMS (*Relational Database managemnt System*) yang bekerja secara cepat dan mudah digunakan. *MySQL* ini yaitu program pengakses jaringan sehingga dapat berguna untuk membangun sebuah aplikasi yang multiuser[17].

2.2.2 UML (Unified Modeling Language)


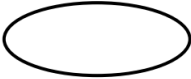

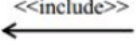
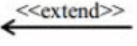
UML (*Unified Modeling Language*) merupakan sebuah visual dalam perancangan sebuah perangkat lunak untuk menguraikan persyaratan, pelaksanaan analisis desain serta menggambarkan *arsitektur pemrogramn*. UML bertujuan untuk memudahkan pengembangan diharapkan mampu mempermudah pengembangan *software* dan mampu memudahkan pengguna dalam memenuhi kebutuhannya secara efektif, lengkap, dan tepat [18]. Dalam proses pengerjaan suatu sistem terdapat beberapa diagram uml yang sering digunakan, seperti *use case diagram*, *activity diagram*, *class diagram* dan *sequence diagram*. Penjelasan lebih detail bisa dilihat dibawah ini:

1) Use case Diagram

Use case diagram adalah diagram yang menggambarkan hubungan antara aktor dengan sistem. *Use case diagram* mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat[18].

Tabel 2. 1 Simbol *Use Case Diagram* [19]



No	Simbol	Nama	Keterangan
----	--------	------	------------



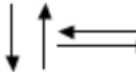
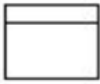
1		<i>Actor</i>	Digunakan untuk merepresentasikan interaksi antara pengguna dan sistem informasi secara jelas dan terstruktur
2		<i>Use Case</i>	Menggambarkan hubungan atau interaksi dengan aktor dalam bentuk abstraksi
3		<i>Association</i>	Menggambarkan secara abstrak hubungan antara aktor dan use case
4		<i>Generalisasi</i>	Merepresentasikan hubungan antara dua use case
5		<i>Include</i>	Menggambarkan bagian dari fungsionalitas <i>use case</i> lainnya
6		<i>Exclude</i>	Memvisualisasikan fungsionalitas Tambahan dengan Kondisi Tertentu

2) Activity Diagram

Activity Diagram merupakan representasi *visual* dari urutan langkah-langkah kerja yang melibatkan aktivitas atau tindakan, termasuk pilihan dan pengulangan. Dengan menggunakan *Activity Diagram* ini akan memudahkan dalam mengembangkan sebuah perangkat lunak dan membantu dalam memahami proses secara keseluruhan [20].

Tabel 2. 2 Simbol *Activity Diagram* [20]





No	Simbol	Nama	Keterangan
1		<i>Initial Node</i>	Simbol untuk menunjukan awal eksekusi suatu proses dalam sistem.
2		<i>Activity</i>	Sistem memulai aktivitas dengan



			tindakan yang diawali oleh kata kerja
3		<i>Decision</i>	Digunakan untuk percabangan aktivitas
4		<i>Final Node</i>	Digunakan untuk mengakhiri objek
5		<i>Line Connector</i>	Garis penghubung dalam Use Case Diagram digunakan untuk menghubungkan satu simbol dengan simbol lainnya secara bersambung
6		<i>Swimlane</i>	Digunakan untuk memisahkan atau mengelompokkan tanggung jawab bisnis terhadap aktivitas yang terkait.

3) *Sequence Diagram*

Sequence diagram merupakan salah satu jenis diagram *UML* yang berfokus pada *visualisasi* interaksi dinamis antara objek atau komponen dalam suatu sistem. Diagram ini menampilkan perilaku skenario sistem dan *entitas* yang saling berinteraksi [21].

Tabel 2.3 Simbol *Diagram Sequence* [21]



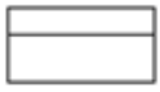


No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Simbol untuk Mewakili entitas eksternal yang berinteraksi dengan sistem, seperti pengguna, sistem lain, atau perangkat eksternal.
2		<i>Entity Class</i>	Simbol untuk memodelkan database
3		<i>Boundary Class</i>	Simbol yang memiliki interaksi dengan aktor
4		<i>Control Class</i>	Simbol yang digunakan untuk menjembatani boundary dan entity

5		<i>A Focus of Control & A Life Line</i>	Simbol untuk memberikan gambaran yang jelas tentang interaksi antar objek
6		<i>A Message</i>	Simbol untuk komunikasi antara objek melalui pengiriman dan penerimaan pesan. Panah menunjukkan arah pengiriman pesan

4) *Class Diagram*

Class Diagram digunakan untuk memetakan rancangan berorientasi objek suatu sistem. Diagram ini memberikan gambaran visual tentang kelas-kelas, atribut, metode, dan hubungan antar kelas. [22].

Tabel 2. 4 Simbol *Class Diagram* [22]

No	Simbol	Nama	Keterangan
1		<i>Generalization</i>	Simbol untuk menggambarkan struktur data dan perilaku dari objek yang ada di objek induk atau ancestor.
2		<i>Nary Association</i>	Simbol untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Class</i>	Simbol untuk mewakili entitas utama dalam sistem, yang memiliki atribut dan metode untuk menyimpan data dan melakukan operasi.
4		<i>Realization</i>	Simbol untuk untuk mengimplementasikan semua metode yang didefinisikan dalam interface.
6		<i>Dependency</i>	Simbol untuk merepresentasikan hubungan "menggunakan" antara dua

			class.
7	————	Assication	Simbol untuk menunjukkan hubungan statis antara dua objek.

2.2.3 Metode SCRUM

Metode *scrum* yaitu sebuah metodologi yang termasuk dalam *agile software development*. *Agile scrum* merupakan metodologi pengembangan perangkat berkembang melalui kolaborasi antar anggota organisasi yang teroganisir. *Scrum* dinilai menjadi *metodologi* pengembangan yang banyak digunakan sekarang karena dapat menciptakan *software* dengan kualitas yang baik, sehingga mampu digunakan dalam pelaksanaan proyek besar ataupun kecil dengan berbagai kemudahan dalam mengadopsi perubahan[23]. Berikut adalah tiga pilar yang mampu memperkuat setiap implementasi dari proses kontrol empiris: [24].

Transparansi : semua orang yang terlibat harus tau secara signifikan terhadap aspek proses yang dijalankan dan semua orang bertanggung jawab atas dampaknya.

- a) *Inspeksi* : Sebaiknya tidak terlalu sering dilakukan karena akan sangat menghambat pekerjaan. Inspeksi dilakukan hanya untuk memastikan setiap artefak scrum dan perkembangan menuju sprint goal tidak terjadi variasi hasil yang tidak diinginkan.
- b) *Adaptasi* : Jika terjadi sesuatu penyimpangan yang mengakibatkan produk yang dijalankan tidak bisa diterima, segera lakukan perubahan proses. Upaya ini dilakukan supaya penyimpangan yang terjadi tidak menjadi begitu parah dan menyimpang terlalu jauh.

Kebutuhan fungsional terdiri dari kebutuhan yang berisikan proses-proses dalam sebuah sistem, pada metode scrum terdiri atas 3 pihak yang terlibat yaitu *Product Owner*, *Scrum Master* dan *Development Team*.

1) *Product Owner*

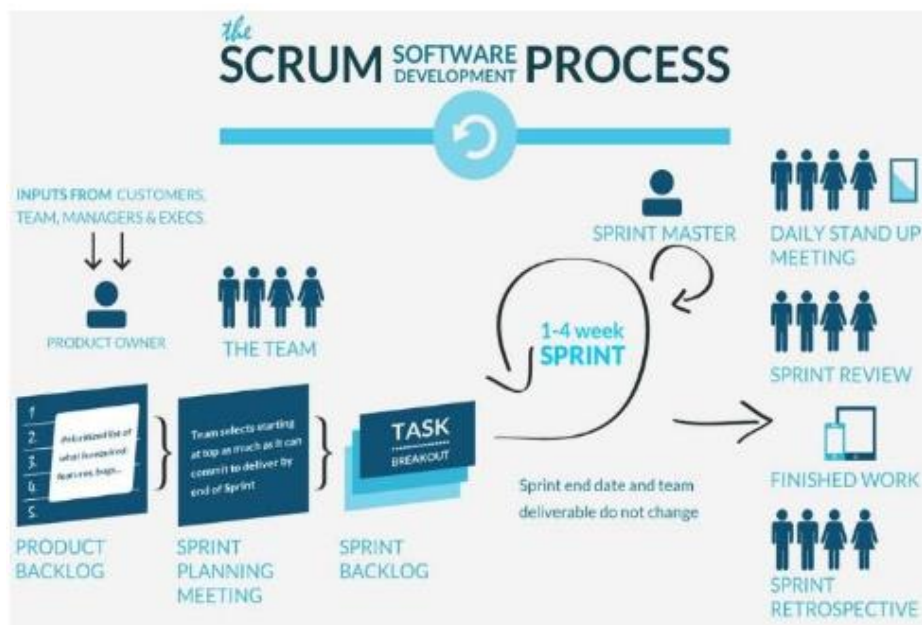
Product owner adalah seseorang yang memiliki tanggung jawab terhadap spesifikasi apa yang akan dilakukan dalam perancangan sistem. *Product owner* bertugas dalam pembuatan *product backlog* yang harus dilaksanakan oleh tim selama proses perancangan sistem[25].

2) *Scrum Master*

Scrum master merupakan *servant leader* yang memastikan semua tim memahami *scrum*. *Scrum master* bertanggung jawab agar proses *development* berjalan dengan baik, serta memastikan tidak ada masalah internal maupun eksternal[26].

3) *Development Team*

Development team mempunyai kewajiban untuk mengerjakan, mengelola, komunikasi yang efektif serta bertanggung jawab sesuai dengan *product backlog* yang diberikan. *Development team* memiliki tanggung jawab terhadap tugas secara keseluruhan [27].



Gambar 2.1 Tahapan Alur Metode SCRUM[34]

Dalam tahap perancangan, scrum memiliki beberapa tahapan rancangan dimulai dari *product backlog*, *sprint planning*, *sprint backlog*, *daily scrum*, *sprint review*, *sprint retrospective*. Berikut adalah tahapan yang ada pada metode scrum:

a) *Product Backlog*

Pada tahapan ini, terjadi pengelompokan kebutuhan pengguna yang diperoleh melalui proses pengumpulan kebutuhan (*backlog* item), yang terdiri atas fitur dan produk mulai dari skala kecil hingga skala besar. Prioritas *backlog* akan mempengaruhi lamanya durasi pengerjaan *sprint*. Semakin tinggi tingkat kerumitan *backlog* maka akan semakin lama pengerjaannya[28].

b) *Sprint Planning*

Pada tahapan ini *product owner* memberikan urutan prioritas Pada tahapan ini, *product owner* bertugas mengurutkan prioritas *product backlog* yang akan digarap. *Product backlog* dikembangkan dalam sebuah *sprint backlog* yang dikerjakan oleh *development team* setiap *sprint*. Selain itu, dalam tahapan ini juga dilakukan pertimbangan mengenai tingkat kesulitan dari *product backlog*, teknologi yang dipakai serta ketentuan iterasi durasi *sprint* [29].

c) *Sprint Backlog*

Sprint backlog merupakan sekumpulan daftar *backlog* yang dipilih sesuai kebutuhan supaya bisa mencapai *sprint goal*. Setiap anggota diatur untuk dapat mengerjakan berbagai bagian yang ada dalam sebuah daftar *sprint backlog*. *Product backlog* diperlukan untuk memenuhi kebutuhan dalam penghematan waktu dan kejelasan pekerjaan tim[30].

d) *Daily Scrum*

Daily scrum merupakan aktivitas harian dalam melakukan proses *sprint* untuk memeriksa produk yang sedang dibuat untuk memeriksa apakah ada hambatan dalam pembuatan produk. *Daily scrum* diperlukan untuk memudahkan dalam melakukan pengecekan dan perbaikan setiap harinya[31].

e) *Sprint Review*

Pada tahapan ini dilakukan guna untuk melakukan inspeksi *increment* dan meninjau apa saja *product backlog* yang sudah dikerjakan maupun belum. Pemeriksaan dan peninjauan tidak hanya berfokus pada hasil pengerjaan, tetapi juga menilai performa kinerja tim sehingga bisa dilakukan perbaikan dalam pengerjaan *sprint* berikutnya[32].

f) *Sprint Retrospective*

Tahap ini berada ketika sprint berakhir. Tahapan ini berisikan penyampaian pendapat dan hasil penilaian kinerja selama penerapan metode scrum berlangsung [33].

2.2.4 Framework SOSTAC Model

Dalam proses memasarkan produk perlu adanya perencanaan dan pengembangan pada aktivitas pemasaran untuk menciptakan, mengkomunikasikan dan mengantarkan *value* produk agar sampai ke audiens[34]. Smith & Chaffey 2018 mengembangkan strategi komunikasi pemasaran bernama *SOSTAC model* yang memiliki 6 tahap perencanaan menggunakan media digital yakni[35]:



Gambar 2.2 Kerangka Kerja SOSTAC Model[10]

- 1) *Situation Analysis* (analisis situasi) merupakan tahap dimana untuk mengetahui situasi atau gambaran suatu perusahaan dari kegiatan apa yang dilakukan, bagaimana interaksi dengan audiens dan pemasaran.
- 2) *Objective* merupakan tahapan dimana perusahaan perlu menetapkan tujuan dalam proses pemasaran yang dilakukan. Bisa menggunakan analisis 5s yaitu sell, serve, sizzle, speak, save.
- 3) *Strategy* merupakan tahap analisis bagaimana cara tujuan yang sudah ditetapkan dalam dicapai. Serta menentukan tujuan segmen pasar yang ingin dijangkau.
- 4) *Tactics* merupakan tahap analisa perusahaan menentukan positioning pada market yang diidentifikasi melalui produk, lokasi, harga, dan promosi. Menurut Elliott (2012) *marketing mix* merupakan set variabel untuk pemasaran agar bisa mengontrol suatu perubahan
- 5) *Action* dalam tahap ini, *SOSTAC model* mencakup taktik dan perencanaan biaya yang dikeluarkan dengan objective dan task method. Serta menentukan timeline untuk menjalankan kampanye.
- 6) *Control* merupakan tahap monitoring dan mengukur performa taktik yang sudah diterapkan dan melakukan kontrol pada setiap taktik yang dijalankan supaya tujuan yang ditetapkan sebelumnya dapat tercapai.

2.2.5 Blackbox Testing

Blackbox Testing merupakan metode yang digunakan untuk pengujian dan berfokus pada fungsionalitas tanpa mempertimbangkan struktur internal atau implementasi teknis dari program yang diuji. Metode ini menguji tanpa harus memperhatikan detail *Software* [36]. Dalam melakukan uji coba menggunakan memperoleh hasil yang tidak bias atau acak memperoleh hasil yang pasti. Dikatakan pasti' menjadi Terbukti Maksudnya jika terjadi kesalahan, maka akan ditolak oleh sistem. Informasi atau data input tersebut tidak bisa diterima. disimpan dalam pangkalan data, sedangkan jika data jika masukan itu tepat, maka dapat diterima atau dimasukkan. sistem informasi basis data[37].