

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan penelitian yang dilakukan Tommi Suryanto (2018) dengan judul “Penerapan *Marketplace* pada *Distro Silver Squad*”, Peneliti melakukan penelitian pada *Distro Silver Squad* untuk membantu dalam proses pelayanan, yang tujuannya mempermudah dalam proses transaksi. Pada penelitian ini, metode yang peneliti gunakan dalam penelitian serta pengembangan adalah metode *Research and Development*. Teknik dalam pengumpulan data yang tepat dalam implementasi lapangan adalah Observasi, Wawancara, Studi Dokumentasi. Proses perancangan *E-Marketplace* pada *Distro Silver Squad* menggunakan bahasa pemrograman *PHP*, *Database MySQL*, *HTML* dan *script* lainnya yang berfungsi untuk merancang *interface* yang *user friendly*.

Berdasarkan penelitian yang dilakukan Nuraida Wahyuni (2018) dengan judul “Pengenalan Dan Pemanfaatan *Marketplace* Untuk Pelaku Ukm Wilayah Cilegon”. Aktivitas jual beli pelaku usaha masih dilakukan dengan cara konvensional yaitu penjual bertemu langsung dengan pembeli dan Pelaku usaha sudah memahami pentingnya terlibat langsung dalam penjualan online dan sudah mempunyai akun pada *marketplace* yang tersedia.

Berdasarkan penelitian yang dilakukan oleh Hilmi Arija Fachriyan (2019) dengan judul “Aplikasi Model *Marketplace* Dalam E-Agribusiness” Keberhasilan e-marketplace sangat bergantung pada tingkat keterlibatan peserta yang berpartisipasi. Kurangnya kehandalan *marketplace* sering dikritik sebagai salah satu faktor kunci yang mengurangi kepercayaan penjual atau pembeli untuk berpartisipasi. Perusahaan *marketplace* agribisnis tidak cukup untuk hanya menyediakan layanan teknis yang baik seperti transaksi *online*. Pengguna lebih memperhatikan bagaimana sebuah *website-marketplace*

Berdasarkan penelitian dari Muhammad Andrean (2017) dengan judul “Perancangan Dan Implementasi Sistem Informasi *Marketplace* Untuk Katering” Penelitian ini menggunakan *System Development Life Cycle* dengan model Structured Design *Waterfall* sebagai skema siklus pengembangan sistem dan menggunakan *Unified Modelling Language* sebagai rancangan pemodelan. Aplikasi dibangun menggunakan Bahasa PHP. Dengan adanya *Marketplace* Catering memudahkan pengusaha catering dalam memasarkan produknya. Serta mempermudah konsumen yang sedang mencari jasa *catering*.

Berdasarkan penelitian dari Robert Marco (2017) dengan judul “Analisis Sistem Informasi *Marketplace* Pada Usaha Kecil Menengah (UKM) Kerajinan Bambu Dusun Brajan” Selama ini, untuk melakukan penjualan masih ada beberapa isu dalam mempromosikan dan menjual produk, sehingga konsumen kurang mengetahui barang apa saja yang diproduksi. Solusi untuk memecahkan masalah di atas, *marketplace* adalah pilihan terbaik untuk mengatasi masalah

penjualan barang kerajinan. Dengan internet, penjual akan memiliki kemudahan dalam mempromosikan dan menjual produk di pasar yang lebih luas.

Berdasarkan penelitian yang dilakukan Tommi Suryanto (2018) dengan judul "Penerapan Marketplace pada Distro Silver Squad", Peneliti melakukan penelitian pada Distro Silver Squad untuk membantu dalam proses pelayanan, yang tujuannya mempermudah dalam proses transaksi. Pada penelitian ini, metode yang peneliti gunakan dalam penelitian serta pengembangan adalah metode Research and Development. Teknik dalam pengumpulan data yang tepat dalam implementasi lapangan adalah Observasi, Wawancara, Studi Dokumentasi. Proses perancangan Marketplace pada Distro Silver Squad menggunakan bahasa pemrograman PHP, Database MySQL, HTML dan script lainnya yang berfungsi untuk merancang interface yang user friendly.

II.2. Landasan Teori

II.2.1. Aplikasi

Pengertian aplikasi adalah suatu bagian dari perangkat lunak yang dapat digunakan untuk menyelesaikan masalah-masalah yang khusus yang dihadapi *user* dengan menggunakan kemampuan komputer. (Fergawan Listianto, dkk : 2017)

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan

salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. (Ari Setiaji, dkk : 2020).

II.2.2. Penjualan

Penjualan merupakan syarat mutlak keberlangsungan suatu usaha, karena dengan penjualan maka akan didapatkan keuntungan. Semakin tinggi penjualan maka keuntungan yang akan didapatpun akan semakin maksimal. Untuk mencapai tujuan ini maka sangat diperlukan usaha-usaha agar konsumen mempunyai daya tarik dan sifat loyal dalam berbelanja disuatu unit usaha. Suatu perusahaan tidak akan berkembang apa bila tidak mampu menjual produk yang dihasilkan, sebaliknya suatu perusahaan mampu untuk terus meningkatkan penjualan maka perusahaan tersebut akan mampu untuk eksis dalam persaingan usaha. (Rogi Gusrizaldi, 2018).

II.2.3. *E-Commerce*

Electronic Commerce (E-Commerce) adalah proses pertukaran produk, jasa, dan informasi atau pembelian, penjualan, melalui jaringan internet. E-Commerce merupakan bagian dari e-business, dimana cakupan e-business lebih luas, tidak hanya sekedar perniagaan tetapi mencakup mitra bisnis. (Wirhan Fahrozi : 2020)

E-Commerce merupakan kegiatan atau aktivitas yang berkaitan dengan proses penjualan atau pembelian produk, jasa ataupun informasi antara

perusahaan, rumah tangga, individu, pemerintah, dan masyarakat atau organisasi swasta lainnya, yang dilakukan dengan memanfaatkan komputer pada media jaringan misalnya internet. Jadi dapat di simpulkan e-commerce merupakan segala bentuk transaksi perdagangan atau perniagaan barang dan jasa baik itu proses pembelian, penjualan, pentransferan, atau pertukaran produk, jasa atau informasi yang merupakan suatu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu yang dilakukan secara elektronik. (Andi Ridho Rachman, 2017).

II.2.4 Jenis-Jenis Transaksi *E-Commerce*

Saat ini ada beberapa jenis *e-commerce* yang dapat dibedakan berdasarkan transaksi yang dilakukan oleh pengguna nya, mulai dari usaha kecil sampai usaha yang besar sekali pun dapat dibedakan berdasarkan transaksi e-commerce nya. Jenis-jenis transaksi *E-Commerce*, yaitu:

1. *Collaborative Commerce (e-commerce)*

Kerja sama secara elektronik antara rekan bisnis. Kerja sama ini biasanya terjadi antara rekan bisnis yang berada pada jalur penyediaan barang (supply chain).

2. *Business to Business (B2B)*

Tipe ini meliputi transaksi antar organisasi yang dilakukan di electronic market.

3. *Business to Consumer (B2C)*

Penjual adalah suatu organisasi dan pembeli adalah individu.

4. *Consumer to Business (C2B)*

Komsumen memberitahukan kebutuhan atas suatu produk atau jasa tertentu, dan para pemasok bersaing untuk menyediakan produk atau jasa tersebut ke konsumen.

5. *Consumer to Consumer (C2C)*

Dimana konsumen menjual secara langsung ke konsumen lain atau mengiklankan jasa pribadi di internet. Dapat juga disebut sebagai pelanggan ke pelanggan, yaitu orang yang menjual produk dan jasa ke satu sama lain. (Andi Ridho Rachman, 2017).

II.2.5. Marketplace

Pasar Elektronik (*marketplace*) adalah segala usaha yang dilakukan untuk melakukan pemasaran suatu produk atau jasa melalui atau menggunakan media Internet. Sedangkan *place* sendiri dalam kamus bahasa Inggris artinya adalah tempat. Disini dapat disimpulkan pengertian dari *Marketplace* adalah tempat untuk melakukan pemasaran produk dan jasa melalui atau menggunakan media Internet. Dimana terdapat penjual dan pembeli yang memiliki kedudukan yang sama (Jenie Sundari : 2017)

Marketplace adalah segala usaha yang dilakukan untuk melakukan pemasaran suatu produk atau jasa melalui media internet atau jaringan *world wide web*. Sedangkan *place* sendiri dalam kamus bahasa Inggris artinya adalah tempat. Jadi, pengertian dari *Marketplace* adalah tempat atau wadah untuk melakukan

pemasaran produk atau jasa melalui atau menggunakan media internet (Agus Dwi Cahya : 2021)

Marketplace merupakan media online berbasis internet (*web-based*) tempat melakukan kegiatan bisnis dan transaksi antara pembeli dan penjual. Pembeli dapat mencari supplier sebanyak mungkin dengan kriteria yang diinginkan, sehingga memperoleh sesuai harga pasar. *Marketplace* adalah wadah komunitas bisnis interaktif secara elektronik yang menyediakan pasar dimana perusahaan dapat ambil andil dalam B2B e-Commerce dan atau kegiatan e-Business lain. Dari beberapa definisi tersebut, dapat dikatakan bahwa marketplace merupakan sebuah wadah pemasaran produk secara elektronik yang mempertemukan banyak penjual dan pembeli untuk saling bertransaksi. (Deni Apriadi : 2017)

Marketplace adalah aplikasi atau situs web yang memberi fasilitas jual beli *online* dari berbagai sumber. Pemilik situs web atau aplikasi tidak memiliki produk apapun dan bisnis mereka hanya menyajikan produk orang lain kepada pengguna kemudian memfasilitasinya. (Iwan Setiawan : 2018)

Marketplace merupakan sebuah wadah online yang menggunakan platform internet (*web-based*) untuk melaksanakan berbagai aktivitas bisnis dan transaksi antara pembeli dan penjual. *Marketplace* merupakan tempat ataupun lokasi yang berbasiskan online di mana pembeli dan penjual melakukan transaksi komersial seperti menjual barang, jasa, maupun informasi. (Ratnasari : 2019)

Marketplace merupakan media online berbasis internet (*web-based*) tempat melakukan kegiatan bisnis dan transaksi antara pembeli dan penjual. Pembeli dapat mencari supplier sebanyak mungkin dengan kriteria yang diinginkan,

sehingga memperoleh sesuai harga pasar. Sedangkan bagi supplier/penjual dapat mengetahui perusahaan-perusahaan yang membutuhkan produk/jasa mereka. (Putra : 2017)

II.2.5.1. Komponen Marketplace

Beberapa komponen yang menunjang sebuah marketplace itu sendiri, yaitu:

1. Pelanggan berasal dari seluruh dunia, yang surf melalui Web.
2. Penjual jutaan toko ada di Web, iklan dan menawarkan barang yang sangat bervariasi.
3. Barang dan jasa mempunyai tipe fisik dan digital. Digital produk ini adalah barang yang diubah menjadi format digital dan di kirim melalui Internet.
4. Infrastruktur Network, hardware, software dan lainnya adalah infrastuktur yang harus disiapkan dalam menjalankan marketplace.
5. Front-end Penjual dan pembeli berhubungan dalam marketplace melalui sebuah front-end. Front-end ini berisi portal penjual, catalog elektronik, shopping cart, mesin pencari, mesin lelang,
6. Back-end Aktivitas yang berhubungan dengan pemesanan dan pemenuhan pemesanan, manajemen persediaan, pembelian dari pemasok, akuntansi dan finansial, proses pembayaran, pengepakan, dan pengiriman dilakukan di back end.

7. Intermediaries Pihak ke tiga yang mengoperasikan antara penjual dan pembeli. Kebanyakan dioperasikan secara komputerisasi (Deni Apriadi : 2017)

II.2.6 *Android*

Android adalah suatu sistem operasi yang berbasis *Linux* dengan perangkat *Mobile device*, *Android* menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi-aplikasi baru untuk membantu kebutuhan manusia. Awal terdirinya *android* dimulai dari *Google Inc* yang membeli *Android inc* pendatang baru dari perangkat lunak untuk ponsel/*smartphone*. Semakin banyak *smartphone* dan *PC Tablet* menggunakan sistem operasi dengan versi yang berbeda. Semakin tinggi versinya maka fiturnya semakin canggih dan banyak. Telepon pertama yang memakai sistem operasi *Android* adalah *HTC Dream* yang dirilis pada tanggal 22 Oktober 2008. Perkembangan sistem *Android* berkembang sesuai versi keluarnya adapun versi *Android* dimulai dari (Muhamad Zaelani ; 2017: 5)

Tabel. II.1 Versi *Android*

Versi	Nama	Rilis	Catatan
1.0	<i>Android 1.0</i>	23 September 2008	<i>Android</i> pertama hanya untuk <i>smartphone</i>
1.1	<i>Android 1.1</i>	9 Februari 2008	
1.5	<i>Cupcake</i>	30 April 2009	Mulai pakai kode nama
1.6	<i>Donut</i>	15 September 2009	
2.0- 2.1	<i>Éclair</i>	26 Oktober 2009 (2.0) 12 Januari 2010 (2.1)	

2.2	<i>Froyo</i>	20 Mei 2010	
2.3	<i>Gingerbread</i>	6 Desember 2010	Digunakan pada <i>smartphone</i> jenis lama
3.0-3.2	<i>Honeycomb</i>	22 Februari 2011 (3.0) 10 Mei 2011 (3.1) 15 Juli 2011 (3.2)	Hanya untuk tablet
4.0	<i>ICS (Ice Cream Sandwich)</i>	19 Oktober 2011	<i>Smartphone</i> dan tablet
4.1-4.3	<i>Jelly Bean</i>	9 Juli 2012 (4.1) 13 November 2012 (4.2) 24 Juli 2013 (4.3)	<i>Update</i> untuk memperbaiki dan menambah <i>fitur</i> pada <i>ICS</i>
4.4	<i>Kit kat</i>	3 September 2013	
5.0	<i>Lollipop</i>	12 november 2014 (5.0) 9 Maret 2018 (5.1)	
6.0	<i>Marshmallow</i>	5 oktober 015	Terdapat <i>daze mode</i> , <i>Do Not Disturb mode</i> , mendukung <i>USB tipe C</i> , mendukung pembacaan <i>fingerprint</i> .
7.0	<i>Nougat</i>	22 Agustus 2018	Multitasking

Sumber: Muhamad Zaelani (2017: 5)

Adapun penjelasan dari versi as android diatas adalah sebagai berikut :

1. *Android 1.0: Astro (Alpha)*

Kedua versi awal *Android* ini mungkin agak asing kamu dengarkan. Pasalnya versi *Android 1.0 Astro (Alpha)* dan *Android 1.1* Walaupun belum menggunakan nama makanan manis, kedua sistem operasi *Android* ini tentu menjadi *pionir*. Pasalnya di sinilah *Android* bermula lewat *smartphone* pertama, *HTC Dream*.

2. *Android 1.1 : Bender (Beta)*

Android 1.1 Bender (Beta) ini belum diluncurkan secara publik untuk kebutuhan komersil. *Platform Android* sendiri pertama kali diluncurkan pada

September 2008 dengan andil Andy Rubin yang saat ini dikenal sebagai Bapak *Android*. Walaupun belum menggunakan nama makanan manis, kedua sistem operasi *Android* ini tentu menjadi *pionir*. Pasalnya di sinilah *Android* bermula lewat *smartphone* pertama, *HTC Dream*. (Muhamad Zaelani ; 2017: 5)

3. *Cupcake*

Mulai dari versi ini, *Android* menggunakan nama makanan manis untuk setiap versi yang diluncurkan. *Android 1.5 Cupcake* sendiri dirilis pada tanggal 30 April 2009 dengan berbagai fitur di sebuah perangkat *smartphone* untuk menggantikan *featured phone* kala itu. (Muhamad Zaelani ; 2017: 5)

4. *Donut*

Tentu pada awal perilisannya, sistem operasi *Android* tetap memiliki banyak *bug* yang pengembangnya perlu mengadakan perbaikan. Hal ini dilakukan pada *Android 1.6 Donut* yang dirilis pada 15 September 2009. Ya artinya belum genap setahun semenjak perilsan *Android 1.5 Cupcake* atau hanya berselang 5 bulan saja. *Android* pun menambahkan beberapa pembaruan, terutama dukungan pada layar *smartphone* yang lebih besar. (Muhamad Zaelani ; 2017: 5)

5. *Eclair*

Sama seperti versi sebelumnya, *Android 2.0 & 2.1 Eclair* masih berfungsi untuk menutupi *bug* yang masih ditemukan pada sistem operasi *mobile* ini. Di samping itu, *Android* juga menambah berbagai fitur di dalamnya.

Mulai dari dukungan *Bluetooth* hingga fitur kamera yang mulai menjadi nilai jual *smartphone* kala itu. *Android 2.0 & 2.1 Eclair* digunakan pada perangkat seperti *HTC Nexus One*. (Muhamad Zaelani ; 2017: 5)

6. *Android 2.2 : Froyo (Frozen Yoghurt)*

Mulai versi ini *Android* tampaknya sudah mulai dikenal luas oleh berbagai *brand smartphone*. *Android 2.2 Frozen Yoghurt* alias *Froyo* ini dirilis pertama kali pada tanggal 20 Mei 2010. Walaupun sudah mulai dipergunakan pada beberapa *brand*, namun tetap saja *Android* masih kalah bersaing dengan *Symbian* yang mendominasi pasar *featured phone*. *Android 2.2 Froyo* memberikan peningkatan pada kecepatan kerja, fitur *USB tethering*, *WiFi hotspot*, serta fitur keamanan. (Muhamad Zaelani ; 2017: 5)

7. *Gingerbread*

Belum setahun berselang lagi, *Android 2.3 Gingerbread* kembali diluncurkan pada Desember 2010 dengan berbagai peningkatan yang cukup signifikan. Hal ini terutama pada tampilan tatap muka alias *user interface* yang digunakan. Mulai versi ini, banyak *brand smartphone* mulai melirik menggunakan sistem operasi *Android*. Salah satunya *Samsung Galaxy Series* yang populer hingga saat ini. (Muhamad Zaelani ; 2017: 5)

8. *Honeycomb*

Untuk para pengguna *smartphone* mungkin akan agak asing dengan versi *Android* yang satu ini. *Android 3.0 & 3.2 Honeycomb* yang menggunakan ikon lebah ini memang diperuntukkan penggunaannya khusus untuk perangkat *tablet*. Tentu perilsan *Android 3.0 & 3.2 Honeycomb* pada 10 Mei 2011 ini untuk mendukung *Samsung* yang mulai merilis perangkat *tablet*, *Samsung Galaxy Tab Series* untuk menyaingi *Apple iPad*. (Muhamad Zaelani ; 2017: 5)

9. *Ice Cream Sandwich (ICS)*

Selain itu, *Android* pun juga merilis versi *Android 4.0 Ice Cream Sandwich* yang kembali diperuntukkan untuk perangkat *smartphone*. *Android 4.0 Ice Cream Sandwich* sendiri dirilis pada 19 Oktober 2011 silam. Punya versi nama paling panjang hingga saat ini, *Android 4.0 Ice Cream Sandwich* memberikan banyak pembaruan. Mulai dari animasi yang semakin halus, sederhana, dan mudah digunakan. (Muhamad Zaelani ; 2017: 5)

10. *Jelly Bean*

Peningkatan signifikan terasa saat menggunakan *Android 4.1 & 4.3 Jelly Bean*. Sistem operasi ini sendiri pertama kali dirilis pada Juni 2012 dengan membawa sejumlah peningkatan terutama di sektor pengolahan grafis. Dengan begini, tentu *Android 4.1 & 4.3 Jelly Bean* bisa memberikan peningkatan fungsi pada *user interface* dan *teknologi Vsync* yang digunakannya. (Muhamad Zaelani ; 2017: 5)

11. *KitKat*

Menggunakan nama *brand* cemilan terkenal, *Android 4.4 KitKat* pertama kali dirilis pada Oktober 2013. Versi *Android* terbaik ini pun bisa dikatakan menjadi favorit karena mendukung hampir seluruh perangkat *smartphone* di dunia. Hal ini dikarenakan *Android 4.4 KitKat* dapat memberikan optimalisasi yang baik termasuk pada perangkat *smartphone* yang memiliki *spesifikasi* kurang mumpuni alias cukup rendah saat itu. (Muhamad Zaelani ; 2017: 5)

12. *Lollipop*

Mulai beberapa versi ke belakang, *Android* dan *Google* pun mulai secara rutin memperbarui sistem operasi mereka dalam selang waktu setahun. Termasuk *Android* 5.0 & 5.1 *Lollipop* yang dirilis dan diresmikan pada Juni 2014. Bisa dibilang *Android* 5.0 & 5.1 *Lollipop* menjadi *pionir* dibuatnya *smartphone flagship* dengan *spesifikasi* cukup mumpuni. Versi *Android* ini sudah mendukung arsitektur 64-bit yang sudah memungkinkan penggunaan RAM di atas 3GB. Salah satunya *ASUS Zenfone 2* yang sudah mengusung RAM 4GB saat itu. (Muhamad Zaelani ; 2017: 5)

13. *Marshmallow*

Android 6.0 *Marshmallow* menjadi suksesor dari versi *Android* sebelumnya. Sistem operasi ini sendiri pertama kali diperkenalkan pada Mei 2018 dan mulai dirilis pada Oktober 2018 silam. Sistem operasi ini secara jelas memberikan peningkatan pada sistem keamanan dengan dihadirkannya *fingerprint sensor* sebagai sistem keamanan *biometrik* yang digunakan. Selain digunakan untuk mengunci layar, *fingerprint sensor* ini dapat digunakan untuk otentikasi *Google Play Store* dan pembelian dengan menggunakan *Android Pay*. (Muhamad Zaelani ; 2017: 5)

14. *Nougat*

Untuk saat ini, sistem operasi *Android* ini masih digunakan beberapa *smartphone* yang baru dirilis belakangan ini. *Android* 7.0 & 7.1 *Nougat* pertama kali diperkenalkan pada Juni 2018 dengan menampilkan ikon robot *Android* dengan batangan *Nougat*. Sistem operasi *Android* 7.0 & 7.1 *Nougat*

mengalami perubahan dari segi tampilan antarmuka. Selain itu ada juga fitur *splitscreen* untuk membagi tampilan layar untuk dua aplikasi sekaligus. (Muhamad Zaelani ; 2017: 5)

II.2.7. Mobile

Aplikasi mobile adalah sebuah aplikasi yang memungkinkan untuk melakukan mobilitas dengan menggunakan perlengkapan seperti PDA, telepon seluler atau handphone. Dengan menggunakan aplikasi mobile, maka dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, browsing dan lain sebagainya (Surawijaya Surahman : 2017)

II.2.8. Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu - Integrated Development Environment (IDE) untuk pengembangan aplikasi *Android*, berdasarkan IntelliJ IDEA. Konsep yang dimiliki *Android Inc* ternyata menggugah minat *Google* untuk memilikinya. Pada bulan Agustus 2005, Akhirnya *Android Inc* diakuisisi oleh *Google Inc.* seluruh sahamnya dibeli oleh *Google*. Banyak yang memperkirakan nilai pembelian *Android Inc* Oleh *Google* adalah sebesar USD 50 juta. saat itu banyak yang berspekulasi bahwa akuisisi ini adalah langkah awal yang dilakukan *Google* untuk masuk ke pasar mobile phone. Maka dari situlah banyak pengembang sistem maupun software berlomba lomba untuk membuat atau merancang sistem *Android* menggunakan software –

software yang support dengan *Android*, dan sebagai contoh disini kita pengenalan *Android Studio*. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, *Android Studio* menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi *Android*, misalnya:

1. Sistem versi berbasis *Gradle* yang *fleksibel*
2. Emulator yang cepat dan kaya fitur
3. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat *Android*
4. Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
5. Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
6. Alat pengujian dan kerangka kerja yang ekstensif
7. Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah - masalah lain
8. Dukungan C++ dan NDK
9. Dukungan bawaan untuk *Google Cloud Platform*, mempermudah pengintegrasian *Google Cloud Messaging* dan App Engine

Laman ini berisi pengantar dasar fitur-fitur *Android Studio*. Untuk memperoleh rangkuman perubahan terbaru, lihat Catatan Rilis *Android Studio* (Google developer ; 2017).

II.2.8 Basis Data (*Database*)

Basis data atau *database* merupakan koleksi dari data-data yang terorganisir dengan rapi sehingga data dapat dengan mudah disimpan dan dimanipulasi. Kita dapat menjumpai pemanfaatan *database* dalam kehidupan sehari-hari, seperti penggunaan mesin ATM, sistem akademi universitas/sekolah, sistem informasi penjualan. Salah satu tujuan dari *database* adalah memberikan pengguna suatu pandangan abstrak dari data, yaitu sistem menyembunyikan rincian bagaimana data disimpan dan dipelihara. Sistem *database* harus dibuat semudah mungkin untuk dimengerti karena kebanyakan pengguna sistem *database* adalah orang-orang yang kurang terlatih di bidang teknologi. (Ni Kadek Ceryna, dkk : 2018 : 102)

II.2.9. Java

Java adalah salah satu bahasa pemrograman yang berbasis objek secara murni/ Object Oriented Programming (OOP). Semua tipe data diturunkan dari kelas dasar yang disebut Object. Dengan konsep seperti itu, pemrograman dalam hal mendesain, membuat, mengembangkan dan mengalokasi kesalahan sebuah program dengan basis Java, dapat dilakukan secara mudah, cepat, tepat dan lebih terorganisir (Putra Yudha et al., 2018)

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang

terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana. Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (general purpose), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi (Udaksana & Kusaeri, 2018)

II.2.11. UML (*Unified Modelling Language*)

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan Bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku. UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi.

Unified Modeling Language (UML) biasa digunakan untuk :

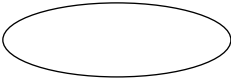
- a. Menggambarkan batasan sistem dan fungsi - fungsi sistem secara umum, dibuat dengan *use case* dan *actor*.
- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.

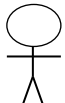
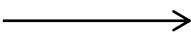
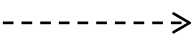
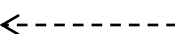
- c. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component and development*.
- f. Menyampaikan atau memperluas *fungsi* dengan *stereotypes*. (Omni Alfina dan Fitriana Harahap : 2019)

II.2.11.1. Use Case Diagram

Use case diagram merupakan pemodelan untuk sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada tabel II.2 di bawah ini :

Tabel II.2. Simbol Use Case

Gambar	Keterangan	Deskripsi
	<i>Use case</i>	Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .

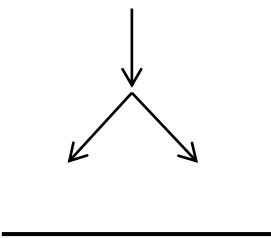
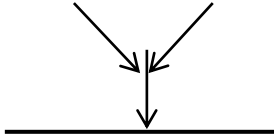
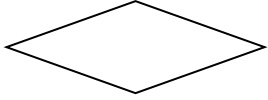
	Aktor	Sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.
	Asosiasi	Penghubung antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi	Penghubung antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i>	Merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i>	Merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Ade Hendini, 2018)

II.2.11.2. Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* dapat di lihat pada tabel II.3 dibawah ini:

Tabel II.3. Simbol Activity Diagram

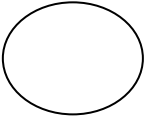
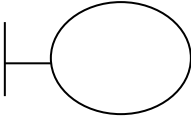
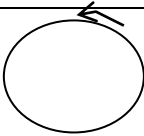
Gambar	Keterangan	Deskripsi
	<i>Start point</i>	Diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i>	Akhir aktifitas.
	<i>Activites</i>	Menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan).	Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	<i>Join</i> (penggabungan)	Digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i>	Menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i>	Pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

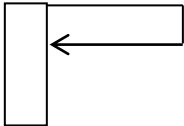
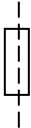
(Sumber : Ade Hendini, 2018 : 109)

II.2..113. *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel II.4 di bawah ini :

Tabel II.4. Simbol *Sequence Diagram*

Gambar	Keterangan	Deskripsi
	<i>Entity Class</i>	Merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i>	Berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i>	Suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek

	<i>Message</i>	Simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i>	Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri
	<i>Activation</i>	Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i>	Garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Hendini, 2018 : 110)

II.2.11.4. Class Diagram

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class* diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class* diagram secara khas meliputi : Kelas (*Class*), Relasi *Associations*, *Generalization* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Ade Hendini, 2018 : 111).

Tabel II.5. *Multiplicity Class Diagram*

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Hendini, 2018)