

## **Bab 2 Landasan Teori**

### **2.1. Perancangan sistem informasi**

#### **2.1.1. Defenisi sistem**

Sistem adalah kumpulan dari beberapa bagian yang memiliki keterkaitan dan saling bekerja sama serta membentuk suatu kesatuan untuk mencapai tujuan dari sisten tersebut. [2]

Terdapat berbagai pendapat yang mendefinisikan definisi sistem, yaitu: “Sistem adalah suatu jaringan kerja dari prosedurprosedur yang saling berhubungan, berkumpul bersamasama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu”, Penjelasan di atas menjelaskan bahwa sistem bekerja dalam suatu jaringan kerja dari suatu prosedur yang saling berhubungan satu sama lain untuk menyelesaikan tujuan dan sasaran yang dimaksud.[3]

#### **2.1.2. Defnisi informasi**

Informasi adalah sebuah data yang diolah menjadi sebuah bentuk yang berguna bagi penerimanya dan bermanfaat dalam mengambil suatu keputusan.[2]

#### **2.1.3. Defnisi sistem informasi**

Sistem informasi adalah sebuah komponen pembentuk sistem yang mempunyai ikatan antara sebuah komponen dengan komponen lainnya yang berfungsi untuk menghasilkan suatu informasi dalam satu bidang tertentu. Hal tersebut berguna untuk membantu perencanaan, pengendalian, dan pengambilan keputusan dengan menyediakan rangkuman rutin dan laporan tertentu. [2]

Definisi sistem informasi menurut Abdul Kadir adalah “sistem informasi adalah kerangka kerja yang mengkoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (input) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan.[3]

#### **2.1.4. Definisi perancangan sistem**

Perancangan adalah sebuah proses yang merencanakan segala sesuatu lebih dahulu, dan perancangan merupakan suatu wujud visual yang dihasilkan dari bentuk-bentuk kreatif yang direncanakan. Langkah pertama dalam melakukan perancangan desain bermula dari hal-hal yang tidak teratur berupa gagasan atau ide-ide melalui proses penggarapan dan pengelolaan dan menghasilkan hal-hal yang teratur, sehingga hal yang sudah teratur bisa memenuhi kegunaan dan fungsi secara baik.[2]

#### **2.1.5. Definisi absensi**

Absensi adalah suatu pendataan kehadiran, bagian dari pelaporan aktivitas suatu institusi, dan suatu komponen institute itu sendiri yang berisi data-data kehadiran yang disusun dan diatur sedemikian rupa sehingga mudah untuk dicari dan dipergunakan apabila sewaktu-waktu diperlukan oleh pihak yang berkepentingan. Absensi berarti “tidak hadir”, namun bisa diartikan pula bahwa absensi merupakan ketidakhadiran atau kehadiran seseorang, Dimana orang tersebut terlibat dalam suatu organisasi yang mengharuskan adanya pemberitahuan tentang kehadiran atau ketidakhadirannya dalam ruang lingkup organisasi tersebut.”Absensi sangat berpengaruh pada kinerja seseorang serta instansi dimana ia bekerja, yang dapat dijadikan sebagai pertimbangan terhadap tindak lanjut serta pembuatan keputusan bagi kelangsungan perkembangan instansi tersebut. [4]

##### **2.1.5.1. Tujuan absensi**

Absensi bertujuan untuk meningkatkan kinerja dan kedisiplinan karyawan. Daftar absensi sangat penting bagi pimpinan untuk mengetahui data kehadiran karyawannya. Adapun tujuan dari absensi adalah sebagai berikut :

1. Untuk melihat kehadiran karyawan
2. Melihat kedisiplinan karyawan
3. Untuk meningkatkan semangat kerja karyawan
4. Mengetahui keadaan karyawan dalam bekerja

Dengan diterapkannya absensi ini akan membantu meningkatkan mutu dari instansi itu sendiri. Dengan penggunaan sistem absensi ini maka orang-orang akan menilai adanya kedisiplinan atau tanggung jawab dalam melakukan pekerjaan yang baik. Selain itu orang juga akan melihat bahwa sistem absensi yang ada di instansi tersebut berkualitas baik, dengan demikian absensi ini juga akan membantu penilaian yang baik bagi setiap organisasi yang menerapkannya.

#### **2.1.5.2. Sistem pelaksanaan absensi**

Absensi dioperasikan setiap hari kerja pada bagian yang telah disediakan oleh instansi tersebut. Absensi karyawan dipegang oleh seorang karyawan yang bertanggung jawab atas absensi tersebut untuk mengecek dan memperhatikan kehadiran karyawan pada setiap pelaksanaan absensi dan karyawan yang bertanggung jawab atas absensi tersebut akan memberikan laporan kepada pimpinan setiap bulannya.

### **2.2. Jenis-jenis metode dalam perancangan sistem informasi**

Berikut ini adalah merupakan jenis-jenis metode dalam perancangan sistem informasi :

#### a. Metode *Waterfall*

Metode *waterfall* banyak digunakan dalam tahap pengembangan. Metode *waterfall* disebut juga metode tradisional atau metode klasik. Metode *waterfall* disebut juga dengan model sekuensial linier atau siklus klasik. Metodologi air terjun ini menyediakan pendekatan aliran hidup perangkat lunak sekuensial yang dimulai dengan fase analisis, desain, pengkodean, pengujian, dan dukungan.[5]

#### b. Metode *Prototype*

Metode *prototype* adalah sebagai satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, sistem akan berfungsi dalam bentuk yang telah selesai. Dasar dari pemikiran ini adalah membuat prototipe secepat mungkin, bahkan dalam waktu semalam, untuk memperoleh umpan balik dari pengguna yang akan memungkinkan *prototype*

tersebut diperbaiki kembali dengan sangat cepat. Semua perancangan diagram atau model yang dibuat tidak harus sempurna dan final dalam pendekatan *prototype*.

Tujuan utama dari penyiapan perancangan adalah sebagai alat pembantu dalam memberi gambaran sistem seperti materi dan menu yang perlu dimasukkan dalam *prototype* yang akan dikembangkan. Setelah perancangan terbentuk, maka akan dilanjutkan dengan mulai mengembangkan *prototype*. Metode *prototype* sesuai untuk menjelaskan kebutuhan pengguna secara lebih rinci karena pengguna sering mengalami kesulitan dalam penyampaian kebutuhannya secara detail tanpa melihat gambaran yang jelas. Untuk mengantisipasi agar proyek dapat berjalan sesuai dengan rencana, target waktu, dan biaya diawal, maka sebaiknya spesifikasi kebutuhan sistem harus sudah disepakati terlebih dahulu oleh pengembang dengan pengguna dalam hal ini klien.[6]

c. Metode *Rad*

Metode *Rapid application development (RAD)* adalah metode proses pembangunan perangkat lunak yang tergolong dalam teknik *incremental* (bertingkat). Metode *RAD* menekan pada siklus pembangunan pendek, singkat, dan cepat. Waktu yang singkat adalah batasan yang penting untuk model ini. *Rapid application development* menggunakan metode iteratif (berulang) dalam mengembangkan sistem dimana *working* model (model kerja) sistem akan dikonstruksikan di awal tahap pengembangan yang bertujuan untuk menetapkan kebutuhan (*requirement*) pengguna. Model kerja hanya digunakan sesekali saja sebagai basis desain dan implementasi sistem akhir.[7]

d. Metode *Iterative*

Metode *Iterative Incremental* adalah metode pengembangan yang diciptakan untuk menutupi kelemahan dari metode *waterfall*. Metode ini bermula dari suatu proses perencanaan dan berakhir pada proses penempatan, dimana terjadi interaksi

didalamnya. Teknik dari model iterative incremental adalah teknik yang direkomendasikan untuk seluruh aplikasi, mulai dari aplikasi yang kecil hingga aplikasi yang sangat besar, Dengan metode ini setiap aplikasi yang dihasilkan dari suatu iterasi dapat dievaluasi untuk menjadi iterasi selanjutnya. [8]

e. Metode Spiral

Metode Spiral adalah metode proses perangkat lunak yang menggabungkan aspek iteratif model prototipe pengontrol dan aspek sistematis model sekuensial linier dengan elemen baru analisis risiko. Metode ini memiliki empat kegiatan penting:

1. Perencanaan (*planning*).

Tentukan tujuan, alternatif, dan batasan. Pada tahap perencanaan, pembuat memutuskan aplikasi berdasarkan percakapan dengan pemilik toko dan kekurangan toko.

2. Analisis Risiko (*Risk Analysis*).

Identifikasi analisis alternatif dan solusi risiko. Selama fase analisis risiko, Anda dapat menganalisis bagaimana sistem di dalam toko Anda bekerja, apa yang mendukung pengumpulan data, dan menemukan kekurangan dan masalah.

3. Rekayasa (*Engineering*).

Tahap pengembangan produk selanjutnya. Pada tahap engineering, Anda membuat desain dari aplikasi yang sedang Anda bangun

4. Peringkat Pengguna (*Customer Ratings*).

Mengevaluasi hasil rekayasa bentuk spiral memberikan contoh bahwa semakin tinggi jumlah iterasi, semakin lengkap versi perangkat lunak yang dihasilkan.[9]

f. Metode *Fast*

Metode *fast* adalah suatu metode pengembangan sistem yang merupakan kombinasi dari beberapa metode yang sering digunakan, Metode *fast* sangat

membantu dalam pengembangan sebuah sistem karena kerangka kerja yang lebih fleksibel. Dibawah ini adalah fase-fase dari metode *fast* :

a) Definisi Lingkup

Tahap ini adalah tahap pengumpulan informasi yang akan diteliti dan tahap ini dilakukan untuk menemukan inti dari masalah yang terjadi.

b) Analisis permasalahan

Tahap ini adalah tahap menganalisis sebuah permasalahan yang terjadi pada sebuah sistem sebelumnya, Tujuannya adalah untuk menemukan solusi dari masalah tersebut.

c) Analisis Persyaratan

Tahapan ini bertujuan untuk mengurutkan sebuah kepentingan dari kebutuhan bisnis yang bertujuan mengidentifikasi stakeholder, proses dan antarmuka yang diinginkan pengguna dari sistem yang baru.

d) Desain Logis

Merupakan tahap untuk mentransformasikan suatu kebutuhan sistem pada sistem model yang akan dibuat nantinya.

e) Analisis Keputusan

Tahap ini bertujuan untuk memilih perangkat lunak dan perangkat keras yang akan digunakan dalam pengembangan dan implementasi sistem.[4]

f) Testing

Tahapan testing ini dilakukan untuk menguji dan menganalisis setiap fungsi dari sistem yang telah dibuat agar dapat melihat kesalahan-kesalahan yang terjadi.

g) Implementasi merupakan tahapan implementasi aplikasi program yang telah dirancang dan di uji sebagai tahapannya penyelesaian.

### **2.2.1. Metode yang digunakan**

Metode yang digunakan dalam perancangan sistem informasi ini adalah metode *waterfall*. Metode *waterfaal* dipilih karena metode tersebut mudah dalam pengelolaan

dikarenakan hampir semua requirement telah diidentifikasi dan didokumentasikan, tahapannya terurut secara linier, identifikasi dan dokumentasi yang lengkap, dan menyebabkan proses mudah dipahami oleh semua tim yang terlibat. Dalam metode waterfall terdapat beberapa kekurangan dan kelebihan antara lain :

- a. Kelebihan metode waterfall adalah dapat meminimalisir kesalahan yang mungkin terjadi karena tahapan atau fase metode waterfall dilakukan secara terurut atau satu persatu dan tidak dilakukan secara bersamaan.
- b. Kekurangan metode waterfall adalah tidak memungkinkan untuk banyak revisi dan perbaikan dalam prosesnya. Karena aplikasi yang sudah sampai pada tahap pengujian akan sangat sulit kembali lagi seperti mengubah sesuatu yang tidak terdokumentasi dengan baik pada tahap pertama yaitu konsep.

#### **2.2.1.1. Tahapan-tahapan metode *waterfall***

Dibawah ini adalah tahapan-tahapan metode *waterfall* :

- a. Analisa kebutuhan

Analisis kebutuhan merupakan proses pengumpulan persyaratan yang dilakukan secara intensif untuk mengidentifikasi persyaratan dan perangkat lunak untuk memahami jenis perangkat lunak yang dibutuhkan pengguna harus didokumentasikan selama fase ini.[10]

Proses ini berjalan pada sebuah instansi, mulai dari data karyawan, atur jam masuk dan jam pulang, proses absensi masuk dan absensi pulang sampai dengan rekap data absensi. Informasi ini diperoleh melalui wawancara, diskusi, dan survei langsung ke instansi yang dipilih oleh peneliti

- b. Desain proses

Desain proses adalah langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Proses ini bertujuan untuk membuat sebuah

konsep pemodelan system menggunakan UML dan mempresentatitkan user interface dan desain database dengan menggambar ERD dan LRS.

c. Implementasi

Pada tahap ini hasil desain yang telah dibuat dimasukkan kedalam bentuk Bahasa pemograman. Dalam implementasi ini dibuat dengan *framework codeigniter*. [11]

Pengkodean desain harus diartikan kedalam perangkat lunak, hasil dari tahapan ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahapan sebelumnya. [10]

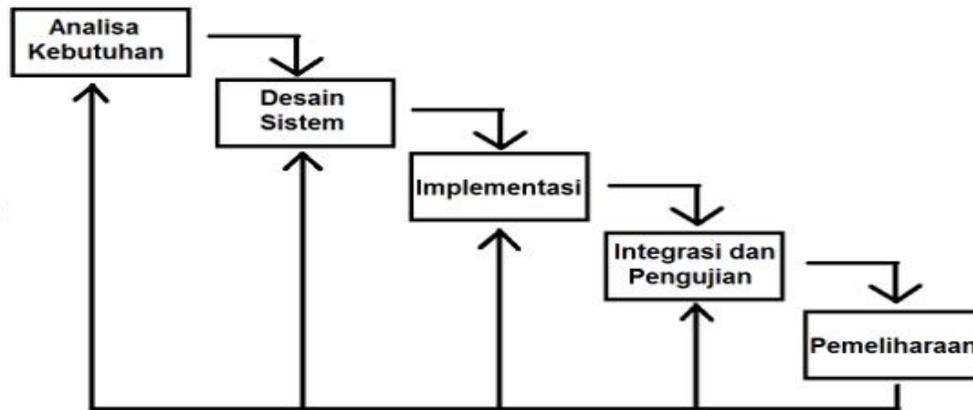
d. Pengujian

Tahap ini adalah tahap pengujian program yang telah dibuat untuk melihat atau memastikan fungsi-fungsi program berjalan sesuai dengan yang diharapkan atau tidak dan hal ini dilakukan untuk meminimalisir kesalahan yang terjadi pada pemodelan system yang telah dibuat.

e. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak mengecualikan kemungkinan bahwa perangkat lunak dapat dimodifikasi saat dikirim ke pengguna karena kesalahan yang tidak terdeteksi yang terjadi selama pengujian atau kebutuhan untuk menyesuaikan perangkat lunak dengan lingkungan baru. Selama fase dukungan dan pemeliharaan, Anda dapat mengulangi proses pengembangan dari awal, menganalisis spesifikasi perubahan perangkat lunak yang ada, tetapi tidak membuat program baru.

Gambar 2.1 adalah bagan metode waterfall yang merupakan metode pengembangan sistem yang digunakan pada penelitian ini. [12]



**Gambar 2.1 – Metode *waterfall***

### **2.3. Uml (*Unified modelling language*)**

UML adalah sebuah metode pemodelan secara visual yang digunakan untuk sarana perancangan sebuah system yang berorientasi objek. Dalam metode UML terdapat 9 jenis diagram yang dapat dikelompokkan berdasarkan sifat statis atau dinamis.[13]

Berikut ini adalah 9 jenis diagram UML :

#### **1. Diagram kelas**

Diagram kelas bersifat statis. Diagram ini menunjukkan satu set kelas, antarmuka, kolaborasi, dan hubungan.

#### **2. Diagram objek**

Diagram objek bersifat statis. Diagram ini menunjukkan objek dan hubungan di antara mereka. Diagram objek menunjukkan contoh statis hal-hal dalam diagram kelas.

#### **3. *Use case* diagram**

Grafik ini statis. Diagram menunjukkan satu set kasus penggunaan dan aktor (jenis kelas khusus). Diagram ini sangat penting untuk mengatur dan memodelkan perilaku sistem yang dibutuhkan dan diharapkan pengguna.

Dibawah ini adalah jenis-jenis symbol dalam *use case* diagram yang akan dijelaskan pada table 2.1 : Sholih . 2006.Pemodelan Sistem Informasi Berorientasi Objek dengan UML.

**Tabel 2.1. Use case diagram**

No	Symbol	Nama	Keterangan
1		<i>actor</i>	Menentukan peran yang dimainkan pengguna saat berinteraksi dengan use case.
2		<i>Generalization</i>	Hubungan di mana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data objek di atas objek induknya ( <i>ancestor</i> ).
3		<i>Use Case</i>	Deskripsi urutan tindakan yang diambil oleh sistem yang mengarah pada hasil yang terukur bagi aktor.

Sumber : Sholih (2006)

#### 4. *Sequence* diagram (Diagram urutan)

Diagram ini bersifat dinamis. Diagram *sequence* merupakan diagram interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu. Penjelasan simbol-simbol yang digunakan pada *sequence diagram* dapat dilihat pada Tabel 2.2.

**Tabel 2.2. Sequence diagram**

No	Simbol	Nama	Keterangan
1		<i>Lifeline</i>	Objek entitas, antarmuka yang saling berinteraksi.

2		<i>Message</i>	Spesifikasi komunikasi antar objek yang berisi informasi tentang aktivitas yang sedang berlangsung.
3		<i>Self-message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
4		<i>Activation</i>	Indikasi dari sebuah objek yang melakukan suatu aksi.

Sumber : Sholiq (2006)

#### 5. *Collaboration* diagram

Grafik ini bersifat dinamis. Diagram kolaborasi adalah diagram interaksi yang menekankan organisasi struktural objek yang mengirim dan menerima pesan.[13]

#### 6. *Statechart* diagram

Grafik ini bersifat dinamis. Diagram ini menunjukkan keadaan sistem, termasuk keadaan, transisi, peristiwa, dan aktivitas. Diagram ini sangat penting untuk menggambarkan sifat dinamis dari antarmuka, kelas, dan kolaborasi, dan sangat penting ketika memodelkan sistem reaktif.

#### 7. *Activity* diagram

Grafik ini bersifat dinamis. Diagram ini adalah jenis khusus dari diagram keadaan yang menunjukkan aliran sistem dari satu aktivitas ke aktivitas lainnya. Diagram ini sangat penting ketika memodelkan fungsionalitas dalam suatu sistem dan menekankan aliran kontrol antar objek. Penjelasan simbol-simbol *activity diagram* dapat dilihat pada Tabel 2.3

**Table 2.3 – Activity diagram**

No	Simbol	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Decision</i>	State dari sistem yang mencerminkan pengambilan keputusan.
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan.
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.
6		<i>Control Flow</i>	Arus aktivitas.
7		<i>Receive</i>	Tanda Penerimaan.

Sumber : Sholiq (2006)

#### 8. Component diagram

Grafik ini statis. Diagram ini menunjukkan konfigurasi dan ketergantungan komponen yang ada. Diagram ini mengacu pada diagram kelas, di mana komponen biasanya memetakan ke satu atau lebih kelas, antarmuka, dan kolaborasi.

#### 9. Deployment diagram

Diagram ini bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (saat run time). Dengan ini memuat simpul – simpul (*node*) beserta komponen – komponen yang ada di dalamnya. Deployment diagram berhubungan erat dengan diagram kompoen dimana deployment diagram memuat satu atau lebih

komponen – komponen. Diagram ini sangat berguna saat aplikasi berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*).

## **2.4.Website**

Website atau situs dapat didefinisikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi tekstual seperti gambar diam atau bergerak, animasi, suara, dan kombinasi dari semuanya. Semua ini bersifat dinamis dan membentuk satu set data bangunan yang saling berhubungan, melalui koneksi jaringan. Hubungan antara halaman web yang satu dengan halaman web yang lain disebut hyperlink, sedangkan teks yang digunakan sebagai media penghubung disebut *hypertext*. [2]

Dalam Website juga terdapat 2 jenis type yaitu :

### 1. Dinamis

Website dinamis adalah isi informasi website selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna website, Sehingga update website bisa dilakukan oleh pengguna maupun pemilik. [14]

### 2. Statis

Website statis adalah dimana isi informasi website tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik website. Artinya, update website hanya bisa dilakukan oleh peggunanya saja.

#### **2.4.1. Bahasa pemograman website**

Bahasa pemrograman website merupakan bahasa komputer yang berisikan suatu proses memberikan intruksi sebuah perintah dalam membuat web sehingga dapat digunakan dan ditampilkan dengan bantuan browse. [15]

Berikut ini adalah jenis pemograman dalam pembuatan sebuah website :

##### a) *Cascading style sheet (CSS)*

Bahasa pemrograman dalam web yang berarti halaman penataan menurun yang menjelaskan tampilan dari sebuah halaman web dalam *mark-up language*

b) *Javascript*

Bahasa pemrograman yang sangat mudah dalam membuat web karena bersifat *clientside programming*. Tipe bahasa yang dilakukan oleh client dalam pembuatannya seperti google chrome dan mozilla firefox.

c) *Hyper text markup language* (HTML)

Bahasa pemrograman struktur standar dalam yang mendeskripsikan sebuah halaman dan berfungsi sebagai publikasi dokumen online kemudian dapat diakses oleh web.

d) *Hyper text preprocessor* (PHP)

Bahasa pemrograman yang dijalankan melalui laman web yang umumnya digunakan mengolah informasi di internet. PHP bersifat gratis dan menyatu dengan HTML tetapi berbeda server dalam script nya.

e) *Bootstrap*

Dalam perancangan sistem atau aplikasi web yang responsive cepat, gratis dan mudah maka digunakannya tools atau framework bootstrap, bootstrap sendiri terdiri dari HTML maupun CSS didalamnya dan dapat dengan mudah menghasilkan atau menampilkan layout, grid, table, form, navigation dan lain-lain. Didalam bootstrap juga terdapat jquery plugins yang dapat menghasilkan UI komponen.

## **2.5. Defenisi *barcode***

Menurut Rotikan (2016:48), “Barcode adalah representasi informasi yang terbentuk dari kombinasi area reflektif tinggi dan rendah pada permukaan suatu objek, yang dapat dibaca oleh mesin yang membagi kombinasi menjadi 1. (satu) dan 0 (nol) dikonversi. Dalam istilah sederhana, kode batang dapat didefinisikan sebagai urutan kode yang terdiri dari bilah dan spasi hitam. Kotak dan spasi hitam memiliki ukuran yang berbeda dan dikelompokkan ke dalam karakter. Dapat mewakili informasi terkait .[16]

Dibawah ini adalah contoh gambar barcode yang akan digunakan pada sistem absensi.



**Gambar 2.2 - Barcode tipe 128**

*Barcode* tidak berisi data deskriptif item, hanya enkripsi numerik. Saat kasir memindai nomor tersebut, kode tersebut secara otomatis ditautkan langsung ke data produk Anda. Hasil pemindai barcode berisi data untuk berbagai produk, antara lain nama pemasok, nama produk, harga, dan data pendukung lainnya. Pemindai kode batang dirancang untuk memproyeksikan data dalam bentuk kode batang yang dipindai.[17]