

# BAB I PERSYARATAN PRODUK

## I.1 Pendahuluan

*Barcode* adalah suatu representasi *data* ke dalam suatu citra (*image*) yang dapat dibaca oleh mesin, citra ini memiliki *data-data* tertentu mengenai suatu produk.

*Barcode* ini banyak digunakan untuk menyimpan informasi mengenai produk yang ditempel *barcode* tersebut. *Barcode* yang ada pada produk-produk ini nantinya akan dibaca oleh *optical reader* untuk menerjemahkan gambar *barcode* menjadi informasi yang dapat diketahui.

*Barcode* ini memiliki banyak jenis dan variasi. Salah satunya adalah *barcode* jenis dua dimensi(2D) yang direpresentasikan dalam bentuk matriks. Dari jenis dua dimensi ini penulis akan membahas *QR Code*.

*QR Code* adalah suatu kode matriks atau *barcode* dua dimensi yang dibuat oleh perusahaan Jepang bernama *Denso-Wave* pada tahun 1994 dengan tujuan konten dari kode tersebut dapat di-*decode* dengan cepat.

Saat ini di negara Korea dan Jepang sudah banyak diaplikasikan *QR Code* untuk berbagai macam keperluan. Beberapa contoh penggunaan *QR Code*, antara lain:

1. Beberapa perusahaan besar menggunakan *QR Code* pada gedungnya untuk menyimpan informasi mengenai perusahaan yang ada di dalam gedung tersebut.

2. Pelabelan pada buah-buahan untuk mengetahui informasi perkebunan yang menghasilkan buah tersebut, pestisida yang digunakan, dan juga pupuk yang digunakan.
3. Penggunaan pada label pasien di rumah sakit untuk memudahkan pembacaan status pasien.

Dengan kelebihan yang dimiliki oleh *QR Code* dan juga makin banyaknya pemilik *gadget* dengan teknologi terkini, maka penggunaan *QR Code* ini dapat memberikan kemudahan-kemudahan antara lain informasi akan lebih cepat diketahui, informasi lebih mudah disimpan, dan lainnya.

### **I.1.1 Tujuan**

Tujuan yang ingin dicapai dalam pelaksanaan tugas akhir ini adalah:

- Membangun aplikasi membaca *file* citra yang berupa *QR Code* dan juga melakukan proses *decode* terhadap *QR Code* sehingga didapatkan informasi yang disimpan di dalam *QR Code* dengan tingkat kesalahan yang sekecil mungkin.
- Membuat fitur *encode* dalam aplikasi sehingga *user* dapat menciptakan gambar *QR Code* yang berisi data informasi. *Data* informasi tersebut dapat di input sendiri oleh *user*.

## **I.1.2 Ruang Lingkup**

Untuk melakukan proses *decoding* terhadap *QR Code version 1-6*, dibutuhkan suatu aplikasi yang dapat mendeteksi gambar dari *QR Code* sehingga didapatkan suatu informasi dari *QR Code* yang sudah dibuat.

## **I.1.3 Definisi, Akronim dan Singkatan**

Berikut adalah akronim dan singkatan yang dapat membantu memperjelas laporan:

- *Software*: Perangkat Lunak
- *Hardware*: Perangkat Keras
- *QR Code*: *Quick Response Code*

## **I.1.4 Overview Laporan**

Sistematika penulisan dalam laporan kerja praktek ini sebagai berikut:

- **BAB I PERSYARATAN PRODUK**  
Berisi uraian garis besar yang meliputi latar belakang, perumusan masalah, tujuan, batasan masalah, sumber *data*, dan sistematika penulisan.
- **BAB II SPESIFIKASI PRODUK**  
Berisi dasar teori yang dipergunakan dalam menyusun laporan kerja praktek ini.

- **BAB III DESAIN PERANGKAT LUNAK**  
Berisi analisi dan perancangan aplikasi yang dibuat meliputi pembuatan proses bisnis dan *flow chart*, *ERD*, *DFD*, dan *Use Case*.
- **BAB IV PENGEMBANGAN SISTEM**  
Berisi penjelasan dari aplikasi bagaimana cara menggunakannya disertai dengan contoh tampilannya.
- **BAB V TESTING DAN EVALUASI**  
Berisi evaluasi tentang aplikasi yang telah jadi.
- **BAB VI KESIMPULAN DAN SARAN**  
Berisi kesimpulan dan saran-saran untuk pengembangan aplikasi ini kedepannya.

## **I.2 Gambar Keseluruhan**

Pembuatan aplikasi yang dapat melakukan membaca *file* citra yang berupa *QR Code* dan juga melakukan proses *decode* terhadap *QR Code* yang sudah dibaca sehingga didapatkan informasi yang disimpan di dalam *QR Code* dengan tingkat kesalahan yang sekecil mungkin ini dilatarbelakangi oleh semakin maraknya penggunaan *Barcode2D* ini di lapangan.

### **I.2.1 Perspektif Produk**

#### **I.2.1.1 Antarmuka Sistem**

Antarmuka aplikasi terbagi menjadi beberapa bagian, antara lain:

- Antarmuka dengan pengguna
- Antarmuka dengan perangkat keras
- Antarmuka dengan perangkat lunak

#### **I.2.1.2 Antarmuka Dengan Pengguna**

Aplikasi ini dibuat dengan memberikan beberapa kemudahan user untuk menggunakan aplikasi ini dan melakukan proses *decode* terhadap kode *QR*.

- Memberikan konfirmasi kesalahan jika *file* gambar yang akan di *decode* tidak berhasil di baca atau sebaliknya

#### **I.2.1.3 Antarmuka Dengan Perangkat Keras**

Perangkat keras yang dibutuhkan agar aplikasi dapat berjalan dengan optimal yaitu: perangkat keras yang terinstalasi pada komputer *User*.

#### **I.2.1.4 Antarmuka Dengan Perangkat Lunak**

Perangkat lunak / software yang digunakan adalah :

- Nama : Microsoft *Visual Studio*  
Versi : 2008
- Nama : *Windows XP Professional Edition*  
Versi : SP2

#### **I.2.1.5 Batasan Memori**

Memori minimum 1 GB. Spesifikasi ini digunakan untuk proses *decode data file* gambar dan penempatan kompilasi.

#### **I.2.1.6 Operasi – Operasi**

Operasi – operasi yang dapat dilakukan pada aplikasi ini adalah :

- *Input file*
- *Decode file*
- *Encode data*

### **I.2.2 Fungsi Produk**

Aplikasi ini memiliki fungsi sebagai berikut :

- *Input file*
- *Decode file*
- *Encode data*

### **I.2.3 Karakteristik Pengguna**

Karakteristik pengguna aplikasi ini adalah :

- Tingkat pendidikan : umum
- Usia : segala usia

## I.2.4 Batasan – Batasan

Batasan – batasan aplikasi antara lain :

Untuk Decoder:

- Versi *QR Code* yang dapat di *decode* adalah versi 1-6
- Ukuran modul piksel gambar adalah 5 x 5.
- Masukkan sistem berupa *file* gambar dengan format *PNG*, *BMP* dan *GIF* dengan ukuran tertentu.
- Kondisi gambar harus sempurna (tanpa ada *blur*, *grayscale*, dll).
- Keluaran sistem berupa pesan teks yang dikodekan ke dalam *QR Code*.

Untuk *Encoder*:

- Untuk fitur *Encode*, Panjang informasi adalah 17 karakter(*QR Code* versi 1) dan *mode* yang didukung adalah *Byte*.

## I.2.5 Asumsi Dan Ketergantungan

Asumsi dan ketergantungan aplikasi ini adalah :

- *Ter-install .NET Framework 2.0*