

BAB II TINJAUAN PUSTAKA

Pada bab tinjauan pustaka akan diuraikan teori dan referensi yang akan menunjang dalam penelitian ini. Teori dan referensi tersebut berkaitan dengan perancangan dan pengembangan produk, *Function Analysis System Technique* (FAST) dan metode *Morphology Chart*, serta beberapa penelitian terdahulu.

2.1 Penelitian Terdahulu

Ada beberapa penelitian sebelumnya yang berhubungan dengan penelitian perancangan sistem ahli perancangan produk.

1. Chunlo (2009) menggambarkan sekelompok system pendukung keputusan (*Group Decisions Suport System/ GDSS*) dikembangkan untuk mendukung lingkungan kerja kolaboratif untuk melaksanakan manajemen nilai (VM) metodologi dalam *briefing* konstruksi. GDSS didasarkan pada metodologi VM yang memungkinkan klien untuk mendefinisikan dan mewakili persyaratan dengan fungsi dan kinerja fungsional, untuk membawa kedepan ide-ide untuk mencapai fungsi, dan akhirnya untuk mengevaluasi, serta menyoroti gagasan terhadap kinerja fungsional untuk pengembangan lebih lanjut dalam desain. Kerangka konseptual, desain, implementasi, penggunaan GDSS dibahas dan dijelaskan dalam urutan itu. Sistem ini dievaluasi melalui studi lapangan dan studi eksperimental. GDSS memiliki potensi besar untuk meningkatkan kinerja penelitian VM dalam hal peningkatan partisipasi dan interaksi, waktu singkat untuk tugas-tugas, dan meningkatkan kepuasan dengan lokakarya VM. Keterbatasan penelitian dan kontribusi pengetahuan penelitian ini juga dibahas.
2. Formentini (2013) menyelidiki bagaimana metodologi yang biasa digunakan dapat mendukung *knowledge transfer* dalam pengaturan multi-proyek secara efektif. Peneliti mengusulkan mengumpulkan pengetahuan dan mentransfer model pengetahuan didasarkan pada teknik analisis nilai, secara empiris dikembangkan dan divalidasi melalui penelitian tindakan dalam industri pembuatan kapal. Model yang diusulkan memfasilitasi pengambilan keputusan di beberapa proyek di desain kapal pesiar dengan mencari penggunaan kembali basis pengetahuan yang tersedia untuk mengidentifikasi solusi desain.

3. Darmawan (2015) telah merancang sebuah *flexible conveyor*, dalam penelitian ini bertujuan untuk membuat desain dan prototipe roller conveyor yang bersifat fleksibel dan memiliki fungsionalitas tinggi. Mudah dalam instalasi dan penyesuaian terhadap bentuk layout produksi yang diinginkan. Selain itu juga harus rendah dan ramah terhadap penggunaan sumber energi listrik, sebagai sumber penggerakannya. Metode pengembangan desain yang dipakai untuk menentukan spesifikasi dari conveyor adalah *Quality Function Deployment* (QFD). Hasil dari QFD akan menjadi masukan bagi pengembangan prototipe conveyor agar sesuai dengan kebutuhan teknis yang diinginkan oleh perusahaan. Sumber energi listrik yang semula dipergunakan untuk menggerakkan mekanisme transfer akan didesain dapat digantikan oleh gaya gravitasi sehingga material dapat berpindah dengan memanfaatkan gaya tersebut, dan hal ini akan menghasilkan penghematan energi listrik yang signifikan serta ramah energi. Dari hasil penelitian terdahulu di atas dapat dijadikan acuan perbedaan dengan penelitian ini yang dapat dilihat pada Tabel 2.1.

Tabel 2.1 Penelitian Terdahulu

No	Nama penulis	Objek Penelitian	Metode	Tools	Hasil penelitian
1.	Chunlo (2009)	Pembuatan Sistem Informasi yang mempermudah fase brainstorming pada proyek Konstruksi di China	GDSS, <i>Value Engineering</i>	VBA for Microsoft Access, C++	Hasil dari penelitian ini ialah sebuah sistem informasi <i>Value Analysis</i> yang menggunakan GDSS untuk mendukung pengambilan keputusan pada saat fase <i>brainstorming</i>
2	Formentini (2013)	Pembuatan Sistem Pengambilan Keputusan pada perancangan Kapal	<i>Value Analysis</i> , DSS, <i>Knowledge Transfer</i>	VBA for Microsoft Excel	Hasil dari penelitian ini ialah terciptanya alat sistem pengambilan keputusan yang dapat digunakan di bermacam macam studi kasus perancangan kapal
3	Darmawan (2015)	Perancangan Flexible Conveyor di Kota Malang	Perancangan Produk	HOQ, QFD	Hasil penelitian berupa <i>flexible conveyor</i> yang bersifat ramah energi dengan menggunakan energi gravitasi sebagai sumber utamanya.
4	Fakhriyudha (2016)	Pembuatan Sistem Ahli pada perancangan produk Conveyor	Diagram FAST, Morphology Chart,	VB. Net	Hasil yang diharapkan adalah terciptanya aplikasi terhadap pengembangan konsep perancangan produk

2.2 Conveyor

Conveyor adalah suatu sistem mekanik yang mempunyai fungsi memindahkan barang dari satu tempat ke tempat yang lain. *Conveyor* banyak dipakai di industri untuk transportasi barang yang jumlahnya sangat banyak dan berkelanjutan. Dalam kondisi tertentu, *conveyor* banyak dipakai karena mempunyai nilai ekonomis dibanding transportasi berat seperti truk dan mobil pengangkut. Jenis *conveyor* membuat penanganan alat berat tersebut/produk lebih mudah dan lebih efektif. Perpindahan tempat tersebut harus mempunyai lokasi yang tetap agar sistem *conveyor* mempunyai nilai ekonomis. Kelemahan sistem ini adalah tidak mempunyai fleksibilitas saat lokasi barang yang dimobilisasi tidak tetap dan jumlah barang yang masuk tidak kontinu. Beberapa jenis *conveyor* ialah *Roller Conveyor*, *Screw Conveyor*, *Bucket Conveyor*, dan *Wire Mesh Conveyor*.

Pada Tabel 2.2 dijelaskan jenis jenis *conveyor* yang sudah ada saat ini. Setiap *conveyor* memiliki kelebihan dan kekurangan masing-masing sehingga setiap *conveyor* memiliki penempatan dan penggunaan yang khusus. *Screw conveyor* yang dapat mengantarkan material dengan bidang vertikal, berbeda dengan *roller conveyor* dan *wire mesh conveyor* yang khusus mengantarkan benda secara horisontal.

Tabel 2.2 Jenis-jenis Conveyor

<i>Roller Conveyor</i>	
<i>Screw Conveyor</i>	
<i>Wire Mesh Conveyor</i>	
<i>Bucket Conveyor</i>	

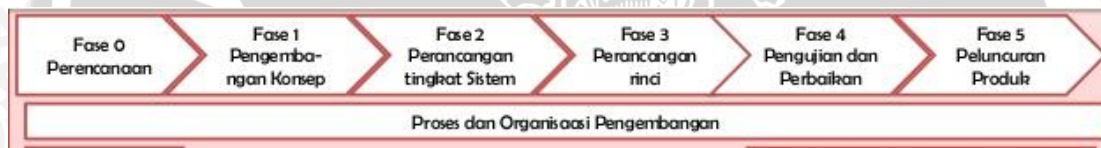
Sumber: Putera (2014)

2.2.1 Flexible Conveyor

Conveyor fleksibel ialah *conveyor* yang didasarkan pada balok – balok *conveyor* yang terbuat dari aluminium atau *stainless steel*, dengan lintasan yang memiliki gaya gesekan rendah. Produk yang ingin di transfer dapat menggunakan palet atau langsung. *Conveyor* ini dapat bekerja di ruang sempit agar menjaga lini produksi mengalir. *Flexible Conveyor* dibuat di sesuai dengan penggunaannya di lapangan, *Flexibel Conveyor* banyak di jumpai pada proses pengemasan makanan, industri farmasi, dan juga di pergudangan toko-toko ritel seperti Wal-Mart dan Kmart.

2.3 Perancangan dan Pengembangan Produk

Pengembangan produk merupakan aktivitas yang diawali dari analisis persepsi dan peluang pasar lalu diakhiri dengan proses produksi, penjualan, dan pengiriman produk. Terdapat enam fase dalam proses pengembangan secara umum menurut Ulrich & Eppinger (2001). Gambar 2.1 adalah fase-fase pengembangan produk.



Gambar 2.1 Fase-fase pengembangan produk.
Sumber: Ulrich & Eppinger (2001).

1. Fase 0 (Perencanaan)

Kegiatan ini disebut sebagai ‘*zerofase*’ karena kegiatan ini mendahului persetujuan proyek dan proses peluncuran pengembangan produk aktual. Didalam fase ini dilakukan penentuan konsep dari pengembangan produk. Proses ini menjadi dasar dari fase-fase berikutnya, maka dari itu sering juga disebut sebagai *zerofase*.

2. Fase 1 (Pengembangan Konsep)

Pada fase pengembangan konsep, kebutuhan pasar target diidentifikasi, alternatif konsep-konsep produk dibangkitkan dan dievaluasi, dan satu atau lebih konsep dipilih untuk pengembangan dan percobaan lebih. Proses yang ada didalamnya ialah analisa kebutuhan dan keinginan pasar serta menyusun alternatif produk dengan variasi dari bentuk, fungsi, dan tampilan suatu produk dan biasanya disertai dengan sekumpulan spesifikasi, analisis produk-produk pesaing serta pertimbangan ekonomis proyek.

3. Fase 2 (Perancangan Tingkatan Sistem)

Fase perancangan tingkatan sistem mencakup definisi arsitektur produk dan uraian produk menjadi subsistem-subsistem serta komponen-komponen. *Output* pada fase ini biasanya mencakup tata letak bentuk produk, spesifikasi secara fungsional dari tiap subsistem produk, serta diagram aliran proses pendahuluan untuk proses rakitan akhir.

4. Fase 3 (Perancangan Detail)

Fase perancangan detail mencakup spesifikasi lengkap dari bentuk, material, dan toleransi-toleransi dari keseluruhan komponen unit pada produk dan identifikasi seluruh komponen standar yang didapat dari pihak ketiga. Rencana proses dinyatakan dan peralatan dirancang untuk tiap komponen yang dibuat dalam sistem produksi. *Output* dari fase ini adalah pencatatan pengendalian untuk produk, gambar untuk tiap komponen produk dan peralatan produksinya, spesifikasi komponen-komponen yang dapat dibeli, serta proses produksi dari produk itu sendiri.

5. Fase 4 (Pengujian dan Perbaikan)

Fase pengujian dan perbaikan melibatkan konstruksi dan evaluasi dari bermacam-macam versi produksi awal produk. Prototipe awal (*alpha*) biasanya dibuat dengan menggunakan komponen-komponen dengan bentuk dan jenis material pada produksi sesungguhnya. Sasaran dari *prototype beta* biasanya adalah untuk menjawab pertanyaan mengenai kinerja dan keandalan dalam rangka mengidentifikasi kebutuhan perubahan-perubahan secara teknik untuk produk akhir.

6. Fase 5 (Produksi Awal)

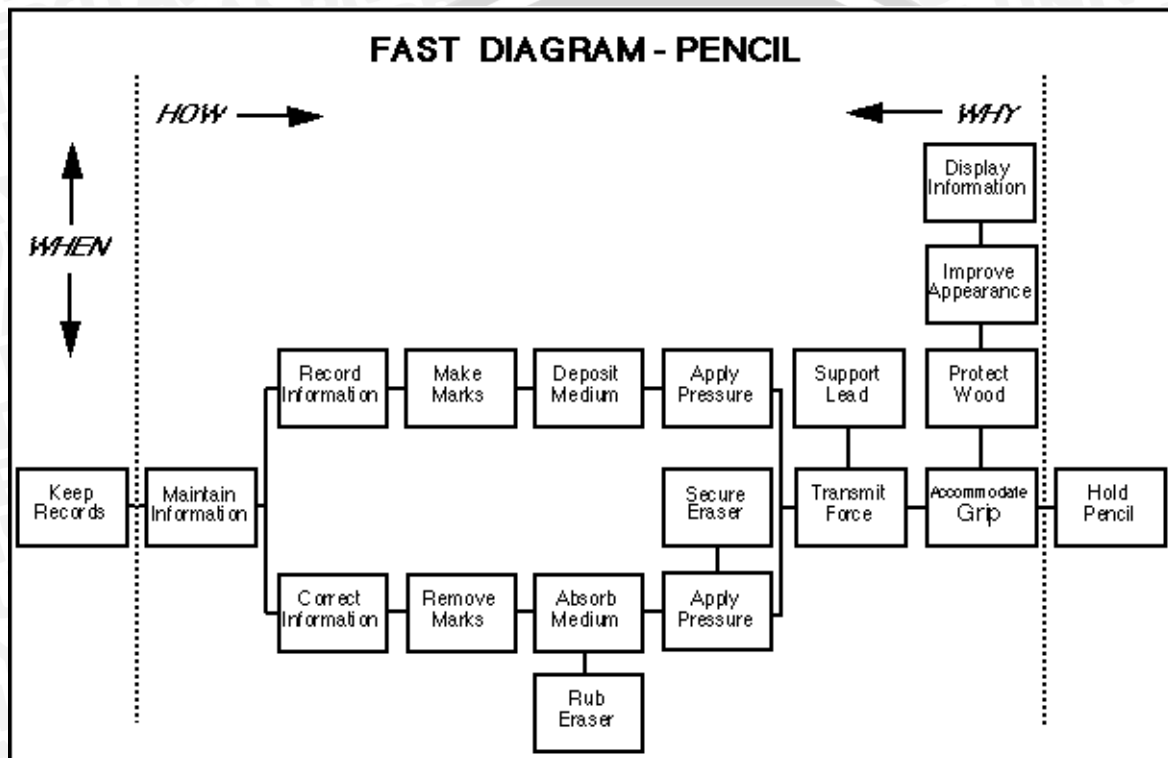
Pada fase produksi awal, produk dibuat dengan menggunakan sistem produksi yang sesungguhnya. Tujuan dari produk awal ini adalah untuk melatih tenaga kerja dalam memecahkan permasalahan yang mungkin timbul pada proses produksi sesungguhnya. Pada beberapa titik pada masa peralihan ini, produk diluncurkan dan mulai disediakan untuk didistribusikan (Ulrich & Eppinger, 2001).

2.4 Diagram FAST

Function Analysis System Technique atau yang biasa disebut dengan diagram *FAST* adalah metode yang digunakan untuk mengidentifikasi fungsi produk dalam *value analysis*. Metode ini dikembangkan oleh Charles W. Bytheway pada tahun 1963. Terdapat dua tipe diagram *FAST* yaitu diagram *FAST* teknik dan diagram *FAST costume*. Diagram *FAST* teknik digunakan untuk yang digunakan untuk mengetahui aspek teknik bagian spesifik dari total produk, sedangkan diagram *FAST costumer* digunakan untuk

mengetahui aspek yang diinginkan konsumen namun tidak digali secara mekanis, teknis, maupun fisik produk tersebut.

Basic FAST Model diawali dengan penentuan tujuan sistem dan melakukan *brainstorming* tentang bagaimana tujuan tersebut dicapai dengan baik. Setelah tujuan dapat diidentifikasi, kemudian tujuan tersebut dituliskan di sisi kiri model, yang nantinya disisi kanan akan menjadi jawaban bagaimana memperoleh hal tersebut. Adapun bentuk diagram *basic FAST model* sesuai dengan Gambar 2.2.



Gambar 2.2 FAST Diagram.
Sumber: Ulrich & Eppinger (2001).

Beberapa fungsi pada garis *How* atau *Why* adalah jalur fungsi kritis (*critical path function*). Sedangkan *minor critical path* dapat menggambarkan bagaimana fungsi bebas atau fungsi pendukung dapat diselesaikan. Fungsi bebas terletak diatas garis kritis, dan aktivitas ditempatkan di bawah garis kritis. Garis vertikal dari fungsi merupakan arah *When*. Fungsi ini terjadi bersamaan atau disebabkan oleh fungsi pada garis kritis. Fungsi bebas tidak tergantung pada fungsi lainnya. Langkah – langkah FAST diagram ialah sebagai berikut :

1. Menentukan batasan proses konseptual. Beberapa fungsi mungkin ditemukan menjadi diluar batas keinginan dari desain yang diidentifikasi oleh fungsi tujuan, dan jika menentukannya akan membuang waktu dan sumber daya.

2. Mengidentifikasi fungsi dasar dari konsep desain yang baru. Fungsi dasar merupakan alasan utama untuk konsep yang ada secara fisik. Jika sistem kompleks, akan ada lebih dari satu fungsi dasar. Gunakan *block diagram* yang bernomor untuk menampilkan fungsi.
3. Menggunakan pernyataan logis untuk menguji definisi dari fungsi dasar.

Customer Requirements/ Functions	Importance	Mechanisms					
		Lead	Eraser	Body	Paint	Band	
Make Marks	30	⊙ 150					
Remove Marks	20		⊙ 100				
Prevent Smudges	15	○ 45		○ 45			
Support Lead	5			○ 25			
Improve Appearance	10			○ 30	○ 30	△ 10	
Accomodate Grip	20			⊙ 100	△ 20		
Column weight	555	195	100	200	50	10	
Mech. weight	1.0	.351	.180	.360	.090	.018	
Mech. target cost	2.80	.98	.51	1.01	.25	.05	
Mech. actual cost	2.92	1.20	.43	.94	.10	.25	

⊙ Strong correlation weight factor = 5
 ○ Moderate correlation weight factor = 3
 △ Weak correlation weight factor = 1

Gambar 2.3 Hubungan antara FASTdiagram dan QFD.
Sumber: Ulrich & Eppinger (2001).

Gambar 2.3 menjelaskan tentang hubungan antara FAST diagram dan Quality Function Deployment (QFD), dimana korelasi yang terjadi diantara keduanya menjadi nyata saat memasukan tahapan pengolahan *input* berupa HOQ yang telah dihasilkan oleh QFD. Kombinasi pendekatan QFD dan FAST diagram ini diharapkan dapat menghasilkan produk yang sesuai dari sisi rancangan alat menurut *user* serta adanya peningkatan *value* lebih baik pada *prototyping level*.

2.5 Morphology Chart

Morphology chart adalah suatu daftar atau ringkasan dari analisis perubahan bentuk secara sistematis untuk mengetahui bagaimana bentuk suatu produk dibuat. Di dalam *chart* ini dibuat kombinasi dari berbagai kemungkinan solusi untuk membentuk produk-produk yang berbeda atau bervariasi. Kombinasi yang berbeda dari sub solusi dapat dipilih dari *chart* mungkin dapat menuju solusi baru yang belum teridentifikasi sebelumnya. *Morphological Chart* berisi elemen-elemen, komponen-komponen atau sub-sub solusi yang lengkap yang dapat dikombinasikan. Langkah-langkahnya adalah sebagai berikut:

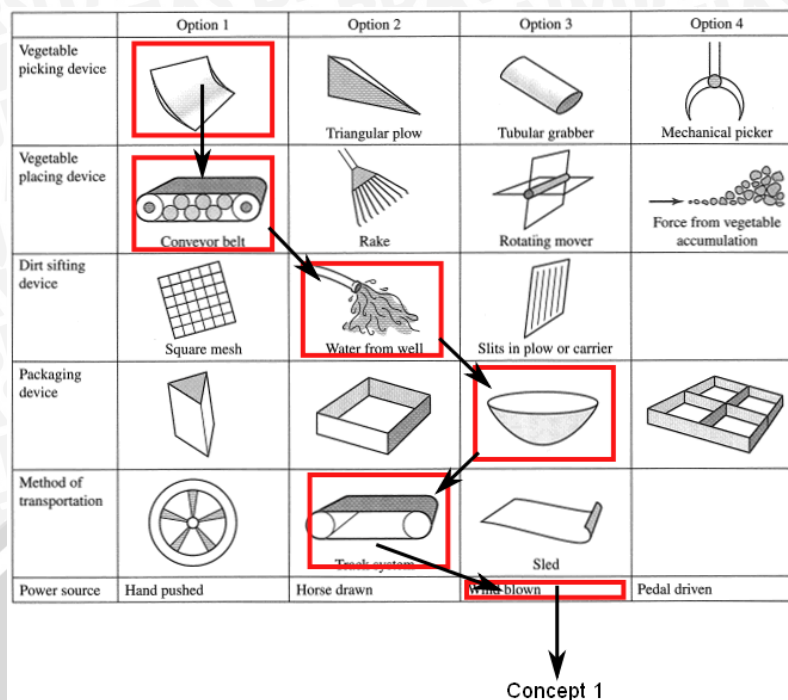
1. Mendaftar/membuat daftar yang penting bagi sebuah produk. Daftar tersebut haruslah meliputi seluruh fungsi pada tingkat generalisasi yang tepat.

2. Mendaftar setiap fungsi yang dapat dicapai yang menentukan komponen apa saja untuk mencapai fungsi. Daftar tersebut meliputi gagasan baru sebagaimana komponen-komponen yang ada dari bagian solusi.
3. Menggambar dan membuat sebuah *chart* untuk mencantumkan semua kemungkinan-kemungkinan hubungan solusi.
4. Identifikasi kelayakan gabungan/kombinasi sub-sub solusi. Jumlah total dari kombinasi tersebut mungkin sangat banyak sehingga pencarian strategi mungkin harus berpedoman pada konstrain atau kriteria.

Pemusatan terhadap suatu solusi (pemecahan masalah) merupakan komponen rancangan yang utama, dimana dari satu sisi hal tersebut dipandang sebagai suatu tindakan yang kreatif dan misterius atau juga sebagai proses yang logis terhadap penyelesaian suatu masalah. Kemungkinan dari suatu rancangan adalah dengan membuat sebuah proposal terlebih dahulu dalam membuat sesuatu yang baru yang belum ada.

Untuk itulah ragam/variasi yang dibuat dianggap sebagai suatu ciri kegiatan yang penting. Hal ini juga merupakan suatu cara dalam mewujudkan dan mengembangkan banyaknya pemikiran yang kreatif terutama suatu kreatifitas seringkali dapat dipandang sebagai suatu penawaran atau penggabungan kembali elemen-elemen kegiatan yang sudah ada.

Tujuan utama dari metode ini adalah untuk memperluas penelitian terhadap solusi baru yang mungkin. Morfologi yang dimaksud adalah mempelajari suatu bentuk atau susunan maka analisis morfologi adalah suatu usaha yang sistematis untuk menganalisis bentuk yang terdapat suatu mesin atau produk, dan grafik morfologi adalah penggambaran secara ringkas dari kesimpulan analisis ini. Seperti apa yang dicontohkan pada Gambar 2.4 merupakan contoh pemilihan konsep untuk alat pemilah sayur.



Gambar 2.4 Contoh *Morphological Chart*.
Sumber: Haik&Shanin (2011).

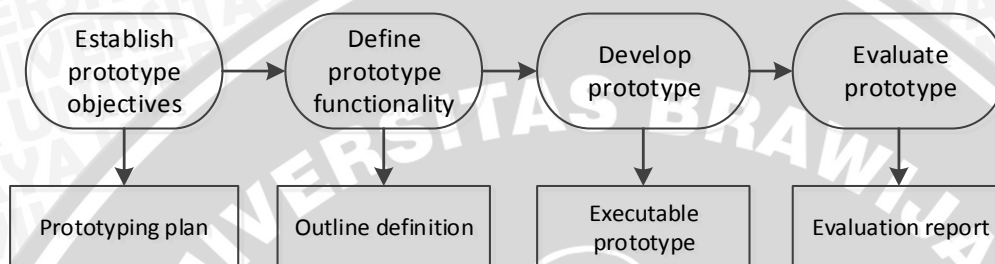
2.6 Prototyping

Menurut Kendall (2003), *prototyping* adalah suatu teknik pengumpulan data yang sangat berguna melengkapi siklus hidup pengembangan sistem tradisional. Saat penganalisis sistem menggunakan *prototyping*, mereka berusaha mencari reaksi, saran – saran, inovasi, rencana revisi pengguna untuk membuat peningkatan terhadap *prototype* sekaligus memodifikasi rencana sistem dengan biaya dan gangguan maksimum.

Istilah *prototyping* memiliki beberapa arti berbeda, dimana empat diantaranya paling banyak digunakan. Definisi *prototype* pertama adalah penyusunan *prototype patch*. Definisi kedua adalah *prototype* operasional yang digunakan untuk menguji fitur – fitur desain tertentu. Konsepsi ketiga adalah menciptakan *prototype* pertama yang benar – benar operasional. Sedangkan jenis *prototype* yang keempat adalah *prototype* fitur – fitur terpilih yang memiliki beberapa fitur – fitur sistem inti. Empat petunjuk utama untuk mengembangkan suatu *prototype* adalah:

1. Bekerja menurut modul – modul dapat dikendalikan
2. Membangun *prototype* dengan cepat
3. Memodifikasi *prototype*
4. Menekankan *interface* pengguna.

Sommerville (2011) menjelaskan bahwa terdapat empat tahapan dalam pengembangan *software* menggunakan *prototyping*. Langkah pertama yaitu menetapkan tujuan *prototype* dengan mengidentifikasi masalah pada sistem yang akan dibuat *prototype*-nya. Langkah kedua yaitu mendefinisikan fungsi *prototype* sehingga sesuai dengan kebutuhan *user*. Langkah ketiga yaitu mengembangkan *prototype* dengan merancang desain logis hingga implementasi konsep yang telah dibuat. Langkah terakhir yaitu evaluasi *prototype* dengan membandingkan hasil *prototype* dan kebutuhan *user*. Langkah-langkah pengembangan tersebut dapat dilihat pada Gambar 2.5.



Gambar 2.5 Model proses pengembangan *prototype*
Sumber: Sommerville (2011)

Berdasarkan Gambar 2.5 model proses pengembangan *prototype* yang lebih rinci dapat dijelaskan sebagai berikut:

1. Menetapkan tujuan *prototype*
 - a. Mengidentifikasi masalah dalam sistem yang sedang berjalan dengan menggunakan analisis PIECES (*Performance-Information-Economy-Control-Efficiency-Services*)
 - b. Menetapkan batasan-batasan (*constraints*) atau ruang lingkup *prototype* yang akan dirancang.
 - c. Menetapkan tujuan dan manfaat dari *prototype*
2. Mendefinisikan fungsi *prototype*

- a. Membuat model kebutuhan sistem (*Requirement Modelling*)

Suatu kebutuhan sistem dinyatakan dalam suatu *checklist* yang disebut SRC (*System Requirement Checklist*). SRC adalah fitur-fitur atau karakteristik yang harus ada dalam sistem informasi untuk memenuhi kebutuhan bisnis (*business requirements*) dan yang dapat diterima pengguna. Model kebutuhan sistem ini digambarkan ke dalam lima kategori umum yaitu *output*, *input*, *process*, *performance*, dan *control*.

- b. Membuat model data (*Data Modelling*)

Pada *data modelling*, sistem yang akan dirancang digambarkan dengan aliran data dan informasi yang dihasilkan dari dan untuk *entity* dalam sistem dengan *Data Flow Diagram* agar nampak jelas.

c. Membuat model proses (*Process Modelling*)

Menggambarkan logika atau aturan bisnis yang dapat dinyatakan dengan *flowchart* atau tabel aturan. Model proses nanti akan berguna dalam membangun algoritma program komputer.

d. Pengembangan Strategi (*Development Strategies*)

Development Strategies atau strategi pengembangan adalah merupakan tahap untuk menggambarkan kegiatan yang tersisa dalam tahap analisis sistem. Tahap ini menjelaskan transisi dari analisa sistem ke sistem desain, *prototyping*, dan pedoman desain sistem diakhiri dengan bagaimana pengembangan perangkat lunak yang dirancang.

3. Mengembangkan *prototype*

a. Langkah Desain

1) Desain *Database Logis*

Konsep model data yang telah terbentuk pada tahap *data modelling* dibawa ke bentuk yang logis. Desain logis terdiri dari daftar entitas untuk pembuatan tabel dan normalisasi tabel.

2) Desain *Database Fisik*

Desain fisik merupakan aktualisasi dari desain logis. Disini *entity* sudah berubah menjadi *table* dengan rancangan bentuk fisik.

3) Desain *User Interface* (UI)

Bertujuan untuk membuat rancangan dari tampilan sistem yang nantinya akan berinteraksi langsung dengan *user* (pengguna). Desain *user interface* harus siap digunakan dan sesuai dengan kebutuhan pengguna.

4) Desain Algoritma

Bertujuan untuk merancang tahapan proses apa saja yang harus dilakukan sehingga *input*, *user interface*, dan *database* menghasilkan *output* yang diharapkan dan dapat ditampilkan, algoritma dapat dinyatakan dengan *flowchart* ataupun *pseudocode*.

b. Implementasi, langkah ini adalah membuat aplikasi pada tingkatan *prototype* dari spesifikasi dan konsep desain yang dirancang dengan melakukan pengembangan *database*, *module* dan *user interface*.

4. Mengevaluasi *prototype*

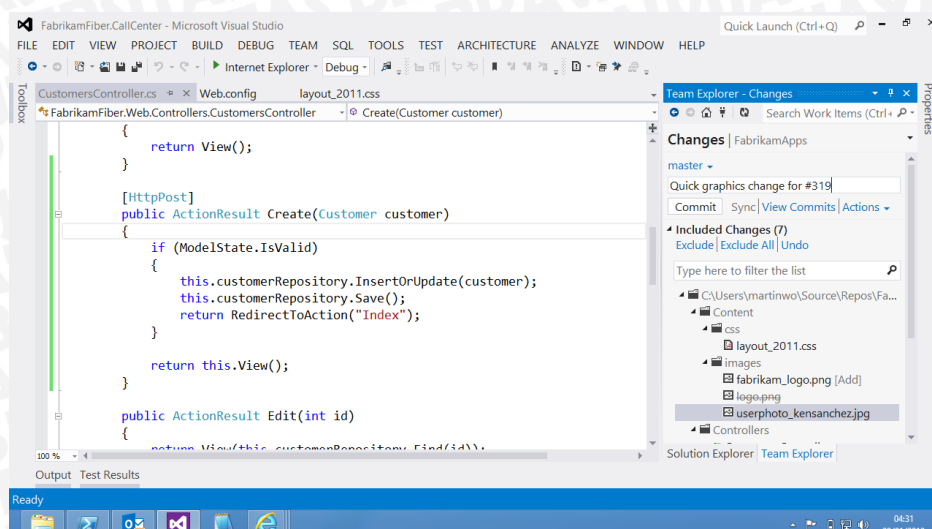
Pada langkah ini dilakukan pengujian terhadap program aplikasi yang telah dibuat. Pengujian program ini ditinjau dari tiga segi, yaitu verifikasi, validasi dan uji *prototype*.

- a. Verifikasi, menguji apakah *prototype* berjalan sesuai yang telah direncanakan. Uji Verifikasi ini meliputi pengujian *hierarki menu*, *form*, *report* dan pengujian ketelitian.
- b. Validasi, menguji apakah fungsi *prototype* yang dirancang telah merepresentasikan kebutuhan user yang meliputi lima kategori umum: *output*, *input*, proses, *performance*, dan *control* (SRC).
- c. Uji *prototype* bertujuan untuk mengetahui apakah *prototype* dapat mengatasi masalah dan kelemahan sistem yang lama

Menurut Wetherbe (1994), untuk mengidentifikasi dan mengevaluasi permasalahan menggunakan *software prototyping*, dapat menggunakan PIECES (*Performance, Information, Economy, Control, Efficiency* dan *Services*).

2.7 Microsoft Visual Basic .NET

Visual Basic adalah salah satu bahasa pemrograman komputer. Bahasa pemrograman adalah perintah perintah yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. Bahasa pemrograman *Visual Basic*, yang dikembangkan oleh *Microsoft* sejak tahun 1991, merupakan pengembangan dari pendahulunya yaitu bahasa pemrograman BASIC (*Beginner's All-purpose Symbolic Instruction Code*) yang dikembangkan pada era 1950-an. *Visual Basic* merupakan salah satu *Development Tool* yaitu alat bantu untuk membuat berbagai macam program komputer, khususnya yang menggunakan sistem operasi *Windows*. *Visual Basic* merupakan salah satu bahasa pemrograman komputer yang mendukung *object* (*Object Oriented Programming* = OOP) (Sempf Bill, 2008).



Gambar 2.6 Tampilan awal Visual Basic.NET
Sumber: Didik (2006).

Menurut Didik Dwi Prasetyo dalam bukunya ‘*Pemrograman Aplikasi Database dengan Visual Basic .Net 2005 dan MS Access*’ menuliskan bahwa *Visual Basic .Net* merupakan salah satu bahasa pemrograman yang bisa digunakan untuk membangun aplikasi-aplikasi .Net di platform *Microsoft .Net*. (2006:1). Dengan menggunakan alat ini, para *programmer* dapat membangun aplikasi *Windows Form*, Aplikasi web berbasis ASP, dan juga aplikasi *command-line*. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti *Microsoft Visual C++*, *Visual C#*, atau *Visual J#*), atau juga dapat diperoleh secara terpadu dalam I. Bahasa *Visual Basic .NET* sendiri menganut paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari *Microsoft Visual Basic* versi sebelumnya yang diimplementasikan di atas *.NET Framework*. Peluncurannya mengundang kontroversi, mengingat banyak sekali perubahan yang dilakukan oleh Microsoft, dan versi baru ini tidak kompatibel dengan versi terdahulu.

2.8 Analisis Masalah

Masih belum adanya alat bantu *material handling* yang berada di PT. Flamboyan Jaya menyebabkan operator harus bekerja lebih untuk memindahkan barang – barang di dalam perusahaan. Seperti dalam proses *blowing* selesai gulungan *plastic* akan diangkut ke mesin *cutting* yang berada pada jarak 4 meter dari mesin *blowing*. Sementara dalam satu *shift* kerja satu mesin *blowing* dapat menghasilkan sekitar 10 hingga 14 gulungan. Oleh karena itu, dibutuhkan suatu alat bantu *material handling* yang sesuai bersifat fleksibel. Dalam penelitian ini, difokuskan pada permasalahan kegagalan inovasi produk ialah berupa 15%

didominasi oleh kurang luasnya pengetahuan yang dimiliki oleh tim R&D dalam menterjemahkan keinginan pasar. Oleh karena itu dibutuhkan suatu program aplikasi untuk membantu tim pengembang dalam perancangan konveyor .

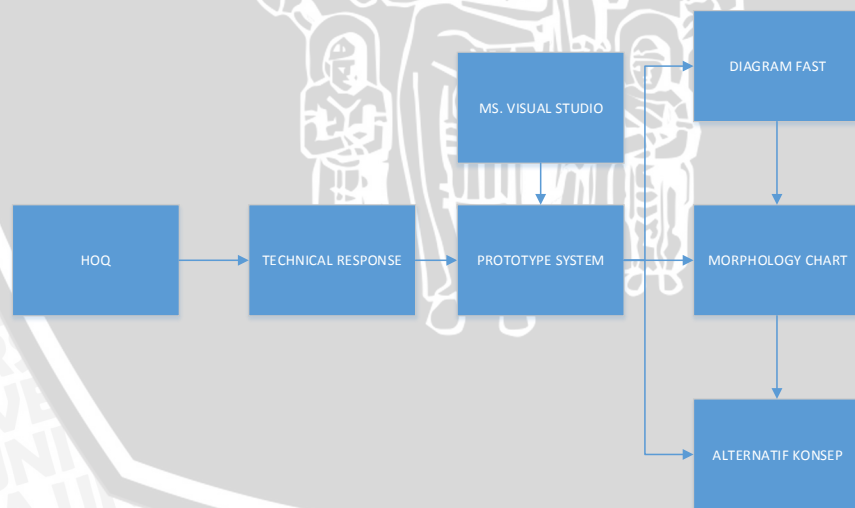
2.8.1 Metode yang Relevan

Berdasarkan studi pustaka pada bab II ini, dapat dipilih metode yang relevan yang akan memberi penyelesaian pada permasalahan yang ada sebagai berikut :

1. Pengumpulan *voice of costumer* dan mengubahnya menjadi QFD agar dapat diketahui *technical response* dari pihak klien akan produk inovasi.
2. Konsep dikembangkan dengan *Morphology chart* dan alternatif konsep dipilih berdasarkan alternatif konsep yang sudah dikembangkan menjadi kombinasi alternatif konsep .
3. Dengan bantuan Microsoft VB.NET akan mempermudah untuk pembangkitan kombinasi alternatif yang dimiliki oleh produk .

2.8.2 Kerangka Solusi

Dari analisis permasalahan dan pemilihan metode yang relevan diatas, maka disusun konsep solusi sebagaimana dalam Gambar 2.7.



Gambar 2.7 Kerangka pemikiran teoritis.