

**DESAIN PERANGKAT LUNAK PADA PROTOTIPE ALAT RAK
PRODUK OTOMATIS UNTUK MINIMARKET**

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Teknik Mesin



Disusun oleh:

Nama : Fariz Rachman Hadi
No.Mahasiswa : 16525033
NIRM : 2016040543

**JURUSAN TEKNIK MESIN
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2021

PERNYATAAN KEASLIAN

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang pernah diajukan orang lain untuk memperoleh gelar sarjana di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah dituliskan atau diterbitkan orang lain kecuali secara tertulis diacu dalam penulisan naskah ini dan disebutkan sebagai referensi. Apabila kemudian hari terbukti pernyataan ini tidak benar, saya sanggup menerima hukum atau sanksi sesuai hukum yang berlaku.

Yogyakarta, Agustus 2021

Penulis



Fariz Rachman hadi

16525033

LEMBAR PENGESAHAN

DESAIN PERANGKAT LUNAK PADA PROTOTIPE ALAT RAK PRODUK OTOMATIS UNTUK MINIMARKET

TUGAS AKHIR

Disusun oleh:

Nama : Fariz Rachman Hadi
No.Mahasiswa : 16525033
NIRM : 2016040543

Yogyakarta, 18 April 2021

Dosen Pembimbing



Mohammad Faizun, S.T., M.Eng., Ph.D.
NIP. 115250101

LEMBAR PENGESAHAN DOSEN PENGUJI

DESAIN PERANGKAT LUNAK PADA PROTOTIPE ALAT RAK PRODUK OTOMATIS UNTUK MINIMARKET

TUGAS AKHIR

Disusun Oleh:

Nama : Fariz Rachman Hadi

No.Mahasiswa : 16525033

NIRM : 2016040543

Tim Penguji

Mohammad Faizun, S.T., M.Eng., Ph.D

Ketua



Tanggal: 30/ 07/ 2021

Agung Nugroho Adi, S.T., M.Eng

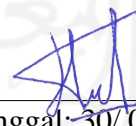
Anggota 1



Tanggal: 30/ 07/ 2021

Donny Suryawan, S.T., M.Eng

Anggota 2



Tanggal: 30/ 07/ 2021

Mengetahui

Ketua Jurusan Teknik Mesin



Dr. Eng. Risdiyono, S.T., M.Eng

HALAMAN PERSEMBAHAN

Segala puji dan syukur kepada Tuhan yang Maha Esa Allah SWT karena dengan izin-Nya dan dengan dukungan serta doa dari orang-orang dan keluarga tercinta, penulis mampu menyelesaikan Tugas Akhir ini. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada:

Orang tua dan saudara-saudara kandung yang selalu mendukung, mendoakan dan menguatkan dalam kondisi apapun sehingga penulis mampu menyelesaikan Tugas Akhir ini dengan baik.

Dosen pembimbing dan dosen-dosen pengajar yang telah meluangkan waktunya untuk membimbing dan memberi arahan selama ini kepada penulis baik dari sisi akademik dan kehidupan.

Rekan-rekan Teknik Mesin UII yang menjadi tempat bertukar pikiran, pandangan dan pengalaman serta saling membantu dalam kebaikan.

HALAMAN MOTTO

"Berusahalah berdiri di atas kakimu sendiri."

(Ibu)

"Bantulah mereka yang membutuhkan bantuan"

(Ibu)



KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Puji syukur penulis panjatkan ke hadirat Allah SWT karena hanya dengan rahmat dan hidayah-Nya, Tugas Akhir ini dapat terselesaikan tanpa halangan berarti. Keberhasilan dalam menyusun laporan Tugas Akhir ini tidak lepas dari bantuan berbagai pihak yang dengan tulus dan ikhlas membantu dan memberikan masukan guna perbaikan Tugas Akhir ini. Oleh karena itu pada kesempatan ini, dengan kerendahan hati penulis mengucapkan terima kasih kepada:

1. Bapak Risdiyono, S.T., M.Eng., Ph.D., selaku Ketua Jurusan Teknik Mesin Fakultas Teknologi Industri Universitas Islam Indonesia,
2. Bapak Mohammad Faizun, S.T., M.Eng., Ph.D selaku dosen pembimbing yang telah memberikan banyak bantuan, bimbingan, serta arahan dalam Tugas Akhir ini,
3. Bapak Purtojo, S.T., M.Sc. selaku dosen pembimbing akademis penulis,
4. Seluruh Dosen dan karyawan di Jurusan Teknik Mesin Fakultas Teknologi Industri Universitas Islam Indonesia, yang tidak bisa disebutkan satu-satu, atas ilmu dan bimbingannya serta bantuannya selama penulis berkuliah di Jurusan Teknik Mesin FTI UII,
5. Keluarga yang selalu mendoakan, mendukung dan menguatkan dalam kondisi apapun.

Penyusunan Tugas Akhir, penulis telah berusaha melakukan yang terbaik namun penulis menyadari Tugas Akhir ini masih jauh dari sempurna. Kritik dan saran yang bersifat membangun sangat penulis harapkan dan dapat ditujukan langsung pada e-mail penulis. Akhir kata penulis mohon maaf yang sebesar-besarnya apabila ada kekeliruan di dalam penulisan Tugas Akhir ini dan penulis berharap Tugas Akhir ini dapat bermanfaat khususnya bagi penulis dan umumnya bagi pembaca.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 18 April 2021

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	ii
LEMBAR PENGESAHAN DOSEN PENGUJI	iv
HALAMAN MOTTO	vi
KATA PENGANTAR	vii
DAFTAR ISI	x
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
Abstrak	xiii
Abstract	xiv
I LATAR BELAKANG	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	2
1.4 Tujuan Perancangan	2
1.5 Manfaat Perancangan	2
1.6 Sistematika Penulisan	3
II TINJAUAN PUSTAKA DAN DASAR TEORI	4
2.1 Tinjauan Pustaka	4
2.2 Landasan Teori	6
2.2.1 Object-Oriented Programming	6
2.2.2 C# Language Programming	6
2.2.3 Software Development Life Cycle	6
2.2.4 Unified Modeling Language	6
2.2.5 Visual Studio 2019	7
2.2.6 Windows Presentation Foundation (WPF)	7

2.2.7	Microsoft SQL Server Management Studio	7
2.2.8	Arduino	8
2.2.9	Komunikasi Serial	8
III METODOLOGI PENELITIAN		9
3.1	Desain Prinsip Kerja Purwarupa Secara Keseluruhan	9
3.1.1	Pembagian Tugas	10
3.2	Alur Desain <i>Software</i>	11
3.2.1	System / Information Engineering and Modelling	11
3.2.2	Software Requirement Analysis	11
3.2.3	Design	12
3.2.4	Coding	12
3.2.5	Testing	12
3.2.6	Maintenance	12
3.3	Peralatan dan Bahan	12
3.4	Analisis Kebutuhan Sistem	13
3.5	Analisis Kebutuhan Perangkat Lunak	13
3.5.1	Kebutuhan Proses	13
3.5.2	Kebutuhan Input	13
3.5.3	Kebutuhan Output	14
3.6	Perancangan Perangkat Lunak	14
3.6.1	Use Case Diagram	14
3.6.2	Use Case Diagram Actors Details	15
3.6.3	Activity Diagram	16
3.7	Rancangan Antarmuka Halaman Setting Port Connection	17
3.7.1	Pemilihan User Interface Elements Setting Port Connection	17
3.8	Rancangan Antarmuka Halaman Proses Pesanan	18
3.8.1	Pemilihan User Interface Elements Item Selections	18
3.9	Metode Pengujian	18
IV HASIL DAN PEMBAHASAN		20
4.1	Pembangunan Sistem	20
4.2	Implementasi Pengujian User Interface	20
4.2.1	Halaman Communication Port	21
4.2.2	Penggunaan Controls WPF Communication Port	23
4.2.3	Halaman Items Selection	23

4.2.4	Penggunaan Controls WPF Items Selection	27
4.3	Implementasi Pengujian Sistem	29
4.3.1	Pengujian User Interface	29
4.3.2	Pengujian Dasar Sistem	31
4.4	Hasil Prototipe Alat	33
4.5	Hasil Pengujian Perangkat lunak dengan Prototipe Alat	34
4.6	Basis Data	35
4.7	Analisis dan Pembahasan	37
4.7.1	Proses Pertukaran Data	37
4.7.2	Synchronous Serial Protocol	37
4.7.3	Siklus Pemecahan Masalah	37
4.7.4	Analisis User Interface	37
4.7.5	Mikrokontroler Proses Pesanan Algoritma	38
4.7.6	Analisis Pengujian Sistem	38
V	KESIMPULAN DAN SARAN	39
5.1	Kesimpulan	39
5.2	Saran	39
	DAFTAR PUSTAKA	40

DAFTAR TABEL

Tabel 4.1	WPF Controls Halaman Communication Port	23
Tabel 4.2	WPF Controls Halaman Items Selection (bagian 1)	27
Tabel 4.3	WPF Controls Halaman Items Selection (bagian 2)	28
Tabel 4.4	Hasil Pengujian User Interface (bagian 1)	29
Tabel 4.5	Hasil Pengujian User Interface (bagian 2)	30
Tabel 4.6	Hasil Pengujian Fungsi (bagian 1)	31
Tabel 4.7	Hasil Pengujian Fungsi (bagian 2)	32
Tabel 4.8	Hasil Pengujian Prototipe Alat	35



DAFTAR GAMBAR

Gambar 3.1 Prinsip Kerja Purwarupa Secara Keseluruhan	9
Gambar 3.2 <i>Software Development Life Cycle (SDLC), waterfall model.</i>	11
Gambar 3.3 <i>Use Case Diagram</i>	14
Gambar 3.4 <i>Paket Data Pesanan</i>	15
Gambar 3.5 <i>Activity diagram</i>	16
Gambar 3.6 Rancangan Antarmuka Pengaturan <i>Port Connection</i>	17
Gambar 3.7 Rancangan Antarmuka Proses Pesanan	18
Gambar 4.1 Halaman <i>Communication Port</i>	21
Gambar 4.2 Dialog box untuk <i>port name</i> kosong	21
Gambar 4.3 Dialog box muncul mengindikasikan koneksi berhasil terba- ngun	22
Gambar 4.4 Halaman <i>Items Selection</i>	24
Gambar 4.5 Satu pesanan berhasil ditambahkan ke dalam list antrian	24
Gambar 4.6 Pesanan lain berhasil ditambahkan ke dalam list antrian	25
Gambar 4.7 Proses setiap pesanan	25
Gambar 4.8 Menghapus seluruh pesanan di antrian	26
Gambar 4.9 Prototipe Tampak Depan	33
Gambar 4.10 Prototipe Tampak Samping	33
Gambar 4.11 Prototipe Tampak Belakang	34
Gambar 4.12 Tabel <i>dbo.Products</i> di basis data	35
Gambar 4.13 Fungsi akses dan update data di basis data (1)	36
Gambar 4.14 Fungsi akses dan update data di basis data (2)	36

Abstrak

Perancangan ini dilakukan untuk membuat perangkat lunak atau aplikasi berbasis desktop menggunakan teknologi Windows Presentation Foundation (WPF) dan bahasa pemrograman C# yang dapat terintegrasi dengan mikrokontroler arduino ATmega 2560. Pada penelitian ini perancangan software dilakukan dengan menggunakan metode Software Development Life Cycle (SDLC) serta menggunakan Unified Modelling Language (UML) sebagai pemodelannya.

Penelitian ini berfokus pada proses penghubungan software C# dan mikrokontroler arduino ATmega 2560, proses pembuatan pesanan, proses pengiriman data pesanan ke mikrokontroler dan mendapat feedback ketika pesanan telah selesai diproses oleh mikrokontroler arduino ATmega 2560 .

Kata kunci : Perangkat lunak, *Windows Presentation Foundation* (WPF), Bahasa pemrograman C#, *Software Development Life Cycle* (SDLC), Komunikasi Serial, *Unified Modelling Language* (UML), Mikrokontroler Arduino ATmega 2560.

Abstract

This design was created to build software or desktop-based application using Windows Presentation Foundation (WPF) technology and C# programming language that can be integrated with arduino ATmega 2560 microcontroller. In this research capacity, the design of software was done using Software Development Life Cycle (SDLC) method along with Unified Modelling Language (UML) to create the model.

This research focused on the connection process of C# software and arduino ATmega 2560 microcontroller and getting feedback when each order is done being processed by arduino ATmega 2560 microcontroller.

Keywords : *Software, Windows Presentation Foundation (WPF), C# Programming Language, Software Development (SDLC), Serial Communication, Unified Modelling Language (UML), Arduino ATmega 2560 Microcontroller.*



BAB I

LATAR BELAKANG

1.1 Latar Belakang Masalah

Kebutuhan akan berbagai macam produk mulai dari makanan, minuman, rumah tangga, perlengkapan bayi dan lain sebagainya adalah kebutuhan yang tidak bisa dipisahkan dalam kehidupan masyarakat di manapun termasuk di Indonesia. *Minimarket*, menjadi tempat salah satu tujuan tempat pembelian berbagai macam produk tersebut oleh masyarakat karena lokasi nya yang tersebar di berbagai tempat sehingga mudah untuk dijangkau.

Pandemi yang terjadi pada akhir tahun 2019 disebabkan oleh Virus Corona sehingga disebut sebagai COVID-19 memberikan dampak yang luar biasa ke setiap aspek kehidupan, salah satunya adalah cara melakukan berinteraksi dan bertransaksi untuk memenuhi kebutuhan sehari-hari. Masyarakat harus melakukan *physical distancing* atau jaga jarak disaat beraktivitas, sehingga penyebaran virus dapat diperlambat.

Metode pembelian di *minimarket* masih menggunakan cara konvensional, yaitu pembeli memilih barang-barang kemudian mengantre di depan kasir untuk melakukan pembayaran. Ketika antre di depan kasir maka terjadi konsentrasi para pembeli di dalam *minimarket* yang artinya meningkatkan potensi terpapar virus.

Inovasi desain rak produk otomatis untuk minimarket di harapkan bisa membantu dan memberikan cara baru dalam proses bertransaksi di dalam *minimarket*, sehingga pembeli cukup menggunakan aplikasi, tidak perlu masuk ke dalam *minimarket*, antre dan berkumpul di dalam tempat pembelian serta dapat mengurangi potensi terpapar virus saat berbelanja.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan di atas maka diperoleh rumusan masalah sebagai berikut:

1. Bagaimana merancang desain sistem rak produk otomatis yang memudahkan pembelian di *minimarket* ?

1.3 Batasan Masalah

Berdasarkan rumusan masalah tersebut pada perancangan desain sistem rak otomatis untuk *minimarket* ini akan dilakukan pembatasan masalah sebagai berikut:

1. Mendesain *software* desktop yang dikembangkan dari bahasa pemrograman C# (CSharp) dan dapat berkomunikasi dengan mikrokontroler Arduino ATmega 2560.
2. Aplikasi desktop dapat mengirimkan data pesanan ke mikrokontroler untuk diproses.
3. Aplikasi dapat mengambil data barang yang dipilih untuk setiap pesanan dan menampilkan status pesanan.
4. Pengembangan aplikasi hanya untuk sistem operasi windows.
5. Hanya 4 produk yang digunakan pada *prototype* ini.
6. Pesanan dikerjakan satu per satu.
7. Tidak membahas mengenai sistem pembayaran.

1.4 Tujuan Perancangan

Berdasarkan rumusan masalah di atas, maka perancangan ini dilakukan dengan tujuan membuat aplikasi desktop sebagai *user interface* antara rak produk otomatis dan pengguna untuk melakukan pemesanan pada rak produk otomatis.

1.5 Manfaat Perancangan

Terdapat beberapa manfaat dari proses-proses selama perancangan ini dilakukan, yaitu:

- a. Sebagai alternatif referensi untuk komunitas-komunitas dalam pengembangan sistem otomasi yang berkaitan dengan *minimarket*.
- b. Mengetahui konsep *object-oriented programming* sebagai paradigma untuk membuat aplikasi desktop.

1.6 Sistematika Penulisan

BAB I : PENDAHULUAN

Pada bab ini dijelaskan latar belakang, rumusan masalah, batasan, tujuan, manfaat, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA DAN LANDASAN TEORI

Pada bab ini dijelaskan teori-teori dan penelitian terdahulu yang digunakan sebagai acuan dan dasar dalam penelitian.

BAB III : METODOLOGI PERANCANGAN

Pada bab ini dijelaskan metode yang digunakan dalam perancangan meliputi alur perancangan, tahapan, dan peralatan yang digunakan.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini dijelaskan hasil perancangan dan pembahasannya.

BAB V : KESIMPULAN DAN SARAN

Pada bab ini ditulis kesimpulan akhir dari perancangan dan saran untuk pengembangan perancangan selanjutnya.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Otomasi menjadi salah satu hal penting dalam proses pemenuhan kebutuhan manusia yang populasinya terus meningkat namun di saat yang sama kualitas barang dan keamanan dalam proses pemenuhan kebutuhan harus tetap terjaga. Pembelian di minimarket menjadi salah satu cara termudah untuk membeli kebutuhan, dan karena COVID-19, kegiatan berbelanja yang aman kedepannya menjadi sangat penting [Edwan et al., 2020].

Putri dan Wibowo 2019, melakukan penelitian "*Prototype Smart Minimarket*" mengenai pengembangan *prototype* sistem *minimarket* yang dapat bekerja dengan otomatis. Penelitian yang dilakukan melakukan pengembangan aplikasi mobile dan PC yang dibuat menggunakan Android Studio, Mobile Vision API, dan Visual Studio sebagai *tools* pengembangan aplikasi yang akan terintegrasi dengan *prototype* alat melalui mikrokontroler NodeMCU. Beberapa fitur yang ada pada aplikasi dalam penelitian tersebut adalah melakukan pemesanan melalui *smartphone*, memperbaharui pesanan di basis data, dan aplikasi PC melakukan pengambilan data dari basis data untuk selanjutnya diberikan kepada mikrokontroler untuk memproses pesanan pada *prototype*.

Setiadi dan Setiawan 2017, melalui penelitian "*Pengembangan Aplikasi Penjualan Sparepart Di Bengkel Anugrah Jaya Motor Berbasis Desktop*" mengembangkan aplikasi desktop untuk membantu proses transaksi dalam hal penyimpanan data transaksi penjualan *sparepart* dan jasa *service* sehingga memudahkan kasir melakukan input data, melakukan pengecekan stok barang, dan pembuatan laporan tanpa perhitungan secara manual lagi. Metodologi pengembangan perangkat lunak yang digunakan adalah *Unified Software Development Process (USDP)* dan untuk memodelkan aplikasinya menggunakan *Unified Modelling Language (UML)*.

Edwan et al., 2020, melalui penelitian "*An Intelligent and Automated Approach for Smart Minimarkets*" membuat sebuah desain dan implementasi *prototype smart minimarket* untuk kota-kota besar yang padat penduduk seperti di ibu kota. Pada penelitian tersebut menghasilkan *prototype* yang memungkinkan pembeli untuk mengakses dan memilih produk yang ada di *minimarket* menggunakan aplikasi *mo-*

bile dan melakukan pemesanan. Melalui aplikasi, penyedia jasa juga dapat mengatur *minimarket* dari jarak jauh seperti melakukan pengecekan jumlah barang dan mendapatkan informasi mengenai pendapatan. Penggunaan *artificial-neural-network (ANN) machine-learning* digunakan untuk memprediksi pengeluaran mingguan pembeli.

Karnadi dan Sitohang 2020, melakukan penelitian mengenai pengembangan *shopping cart system*, "*The Utilization of Smart Price Application Based Android*" yang penerapannya untuk di *minimarket*. Sistem ini meliputi aplikasi berbasis Android sebagai alat untuk mengakses *web service* dan MySQL sebagai *database manager*. Sistem ini bertujuan untuk mempermudah pelanggan dalam berbelanja di *minimarket* melalui beberapa fitur pada aplikasi seperti melakukan pemilihan dan dimasukkan kedalam *shopping cart* melalui *scan barcode* yang tertera pada produk-produk dan melihat berapa total berat dan harga dari produk yang telah dipilih.

Anjani et al., 2020 melakukan penelitian "*Rancang Bangun Sistem Minimarket Otomatis Berbasis IoT*". Tujuan penelitian tersebut adalah membuat aplikasi Android dan *prototype* alat. Aplikasi android berfungsi untuk melakukan pengambilan dan pembelian barang yang kemudian dapat berkomunikasi dengan *prototype alat* melalui mikrokontroler arduino mega 2560 dan ESP8266 untuk memproses barang-barang yang dipilih.

Sistem pembelian secara otomatis atau nonkonvensional sudah diterapkan di beberapa perusahaan-perusahaan retailer besar di luar Indonesia seperti Argos di Inggris dan Amazon Go maupun di dalam negeri seperti JakartaNotebook. Argos memiliki *digital concept store* yang memungkinkan pelanggan untuk memilih barang melalui katalog di tampilan layar yang tersedia di dalam toko [Spotless, 2017], melakukan pembayaran dan ambil barang di tahap terkahir. Amazon Go menggunakan konsep toko yang memungkinkan pelanggan untuk berbelanja tanpa antri dan tanpa kasir [Cheng, 2019]. JakartaNotebook merupakan tempat berbelanja yang menerapkan konsep serupa dimana pelanggan ketika datang ke toko dihadapkan pada laptop untuk memilih barang di katalog, buat pesan, menunggu antrian proses, cek barang sesuai atau tidak dan *checkout* untuk pembayaran.

Pembuatan aplikasi atau *software* menggunakan bahasa pemrograman C# karena komunitas C# di seluruh dunia cukup besar mengingat banyak *software* yang berjalan di sistem operasi *windows* ditulis menggunakan C# sehingga membantu penyelesaian masalah saat proses pengembangan. C# dibuat dan dikembangkan oleh Microsoft sehingga *maintenance* bahasa pemrograman terjaga dengan baik.

2.2 Landasan Teori

Beberapa teori yang digunakan sebagai landasan dalam penelitian akan dipaparkan dalam bab ini. Beberapa bagian menjadi landasan utama dalam pembuatan aplikasi.

2.2.1 Object-Oriented Programming

Object-Oriented Programming (OOP) adalah pendekatan terhadap pengembangan perangkat lunak dimana struktur dari sebuah perangkat lunak didasarkan pada objek-objek yang berinteraksi satu sama lain untuk menyelesaikan tugas. Interaksi ini terjadi dalam bentuk pertukaran pesan-pesan antar objek. Dalam merespon pesan, objek dapat melakukan sebuah tindakan atau *method* [Clark, 2011].

2.2.2 C# Language Programming

C# adalah sebuah *modern object-oriented, general-purpose programming*, yang dibuat dan dikembangkan oleh Microsoft bersamaan dengan *.NET platform*. Ada beragam perangkat lunak yang dikembangkan dengan C# dan *.NET platform* seperti: *office application*, aplikasi web, *websites*, aplikasi dekstop, aplikasi *mobile*, *games* dan banyak lagi [Nakov and &, 2013].

2.2.3 Software Development Life Cycle

Software Development Life Cycle (SDLC) merupakan metode yang digunakan untuk pembuatan atau perancangan perangkat lunak. SDLC memiliki beberapa model yang bisa digunakan, umumnya model yang digunakan adalah *waterfall*, *agile* dan *V-model*. Model *waterfall* cocok untuk *project* yang kebutuhannya jelas di awal sehingga tidak membutuhkan perubahan yang cepat, model *agile* cocok untuk *project* yang membutuhkan perubahan yang cepat karena berdasarkan proses iterasi, begitu juga untuk *v-model* [Balaji, 2012], dengan demikian model yang cocok digunakan untuk *project* ini adalah *waterfall* karena kebutuhan sistem sudah jelas di awal dan tidak membutuhkan perubahan yang cepat.

2.2.4 Unified Modeling Language

UML merupakan metode desain yang digunakan dalam perancangan perangkat lunak yang berorientasi pada objek. UML terdiri dari beberapa model yang da-

pat digunakan serta berbentuk tekstual dan grafik atau diagram. Salah satu model yang dapat digunakan untuk membantu proses desain adalah *Use Case Diagram*.

Use Case Diagram membantu dalam mendeskripsikan siapa saja aktor yang akan berinteraksi dengan sistem atau perangkat lunak yang dibuat dan interaksi apa yang terjadi.

2.2.5 Visual Studio 2019

Visual Studio *integrated development environment (IDE)* adalah sebuah *launching pad* yang bisa digunakan untuk edit, *debug*, menulis kode, dan *publish* sebuah aplikasi. Sebuah IDE adalah program yang kaya akan fitur yang bisa digunakan dalam banyak aspek mengenai pengembangan perangkat lunak. Visual Studio meliputi *compilers, code completion tools, graphical designers* dan masih banyak lagi fitur yang memudahkan memudahkan proses pengembangan perangkat lunak [Microsoft, 2019]. Visual Studio digunakan untuk mengembangkan program komputer seperti *website, windows app, web service, dan mobile apps*.

2.2.6 Windows Presentation Foundation (WPF)

Windows Presentation Foundation adalah teknologi utama Microsoft untuk membuat *graphical user interfaces*, bisa berupa tampilan *form* biasa, *document-centric windows*, animasi kartun, video, dan *immersive 3D environments*. WPF adalah sebuah teknologi yang sangat memudahkan dalam hal pembuatan berbagai jenis aplikasi [Nathan, 2010].

2.2.7 Microsoft SQL Server Management Studio

SQL Server Management Studio adalah *management tool* yang umum mengatur *relational databases, analysis services databases, reporting services objects, dan integration services packages* [Badiozamany, 2010]. SQL Server Management Studio (SSMS) merupakan *integrated environment* untuk mengatur infrastruktur SQL apapun. SSMS digunakan untuk mengakses, konfigurasi, mengatur, administer, dan mengembangkan semua komponen SQL Server, Azure SQL Database, dan Azure Synapsis Analytics. SSMS memberikan penggunaan komprehensif yang menggabungkan berbagai *graphical tools* dengan berbagai *script editor* sehingga menyediakan akses ke SQL Server bagi para pengembang dan para pengelola basis data [Microsoft, 2020].

2.2.8 Arduino

Arduino adalah sebuah *open-source platform* yang digunakan untuk mengkonstruksi dan memprogram elektronik. Arduino bisa menerima dan mengirim informasi ke banyak perangkat, dan bahkan melalui internet untuk memberikan perintah ke perangkat elektronik tertentu. Arduino menggunakan *hardware* bernama *arduino uno circuit board* dan program *software* untuk memprogram *arduino board* [Karhe and Patil, 2015].

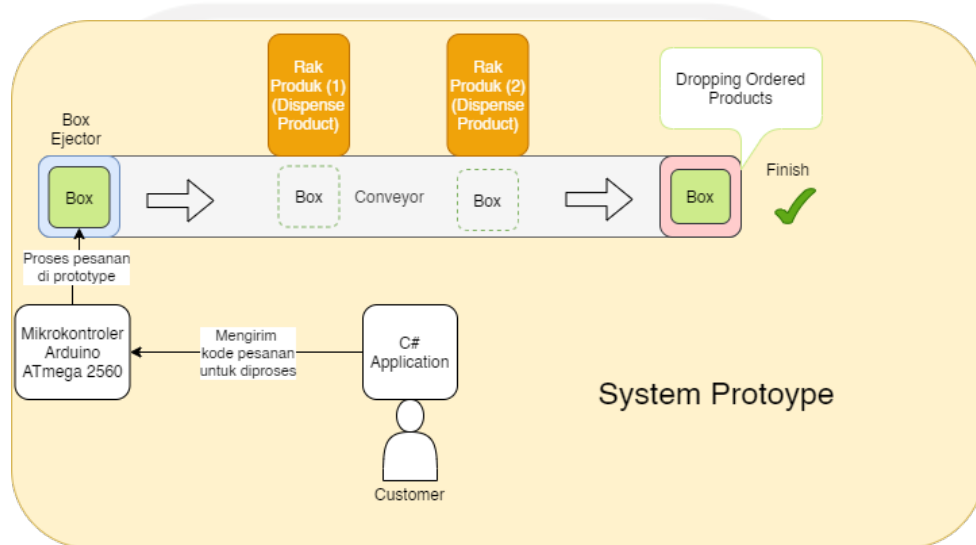
2.2.9 Komunikasi Serial

Secara khusus, komunikasi serial dalam ilmu komputer dan telekomunikasi adalah sebuah tipe komunikasi dimana data dikirimkan satu bit dalam satu waktu pada komunikasi bus [Julien Bayle, 2013]. Salah satu keunggulan menggunakan komunikasi serial adalah membutuhkan lebih sedikit kabel dibanding komunikasi paralel. Dilihat dari sisi kecepatan transfer data, komunikasi serial dengan perkembangan teknologi sekarang mampu memberikan performa yang juga baik jika dibandingkan dengan komunikasi paralel.

BAB III

METODOLOGI PENELITIAN

3.1 Desain Prinsip Kerja Purwarupa Secara Keseluruhan



Gambar 3.1: Prinsip Kerja Purwarupa Secara Keseluruhan

Prinsip kerja purwarupa secara keseluruhan / *big picture* dapat dilihat pada gambar 3.1. Gambar tersebut juga dapat dilihat sebagai alur kerja purwarupa secara garis besar dan proses-proses yang terjadi. Proses-proses yang terjadi secara garis besar melibatkan aplikasi dari C# dan alat purwarupa yang dikontrol oleh mikrokontroler arduino ATmega 2560, penjelasan sebagai berikut:

1. Program C# melakukan pemilihan produk-produk yang diinginkan dan dibuat menjadi pesanan.
2. Program C# mengirim kode pesanan ke mikrokontroler arduino ATmega 2560 untuk diproses.
3. Mikrokontroler arduino ATmega 2560 menerima kode pesanan satu per satu untuk diproses pada prototipe alat.
4. *Box ejector* mendorong box keluar ke konveyor.
5. Box berhenti di rak produk 1 dan 2 untuk mengambil barang-barang yang sesuai pesanan.

6. Setelah semua barang-barang diambil, box masuk ke penampungan barang.

3.1.1 Pembagian Tugas

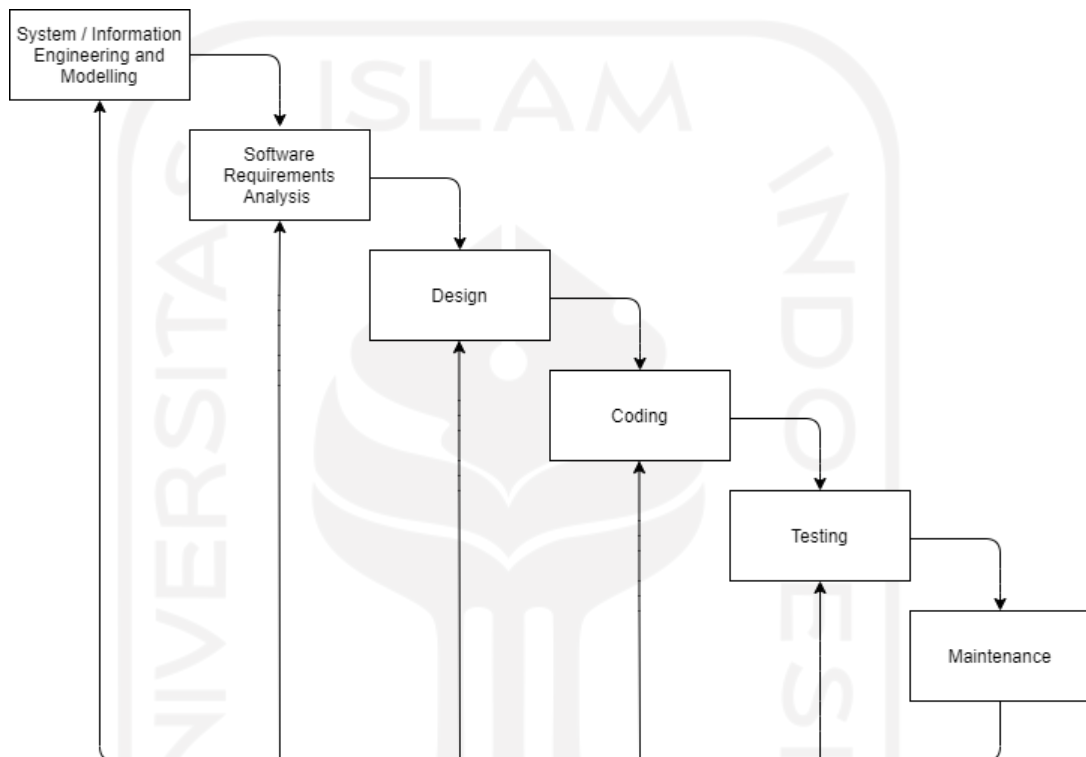
Sistem ini (Perangkat Lunak dan Perangkat Keras Rak Produk Otomatis) secara keseluruhan dikerjakan oleh 3 orang dan pembagian tugas yang dilakukan sebagai berikut:

1. Saya (Fariz Rachman Hadi) melakukan pengembangan perangkat lunak desktop berbasis bahasa pemrograman C# dan teknologi desktop *Windows Presentation Foundation* (WPF) agar dapat melakukan pembuatan pesanan dan pengiriman data ke mikrokontroler Arduino ATmega 2560 serta menerima data *feedback* dari mikrokontroler bahwa pesanan telah selesai.
2. Rois Hendra Purnama (16525105) melakukan desain dan implementasi sistem elektronik yang terlibat pada prototipe alat Rak Produk Otomatis.
3. Ghozali Gaza Rizal Tawakkal (16525054) melakukan desain prototipe alat Rak Produk Otomatis dan fabrikasi prototipe.

Fokus pada tugas akhir ini adalah pembuatan aplikasi menggunakan bahasa pemrograman C# sehingga dapat dilakukan integrasi dan percobaan yang berkaitan dengan pembuatan pesanan pada alat purwarupa yang dikendalikan oleh mikrokontroler arduino ATmega 2560.

3.2 Alur Desain Software

Mengikuti pendapat Roger S.Pressman(2007), tahapan *Software Development Life Cycle* SDLC model *waterfall* untuk pembuatan perangkat lunak dapat dilihat pada gambar 3.2.



Gambar 3.2: *Software Development Life Cycle (SDLC), waterfall model.*

Pada gambar 3.2 dapat dilihat alur penelitian yang dilakukan. Setelahnya akan dijelaskan secara umum setiap tahapan dalam alur penelitian tersebut.

3.2.1 System / Information Engineering and Modelling

Pada tahap ini akan mendefinisikan atau menentukan kebutuhan keseluruhan dari sistem yang akan dibuat kedalam bentuk perangkat lunak. Kebutuhan keseluruhan ini meliputi dari perangkat keras, perangkat lunak serta basis data yang dibutuhkan.

3.2.2 Software Requirement Analysis

Tahap ini membahas mengenai kebutuhan spesifik perangkat lunak yang akan dibuat. Kebutuhan spesifik tersebut meliputi fungsi-fungsi yang dibutuhkan dalam

aplikasi dan tampilan untuk memudahkan penggunaan.

3.2.3 Design

Proses ini mentransformasikan kebutuhan-kebutuhan yang telah ditentukan sebelumnya kedalam bentuk *blueprint* sebagai panduan sebelum proses *coding* dimulai. Proses desain ini dilakukan dengan menggunakan bantuan *Unified Modeling Language* (UML) sebagai metode perancangan perangkat lunak.

3.2.4 Coding

Tahap ini adalah proses penulisan kode berdasarkan desain yang telah dibuat. Proses ini mentransformasikan desain yang ditulis dan dipahami oleh manusia kedalam komputer sehingga komputer dapat mengerti. Penulisan kode ini menggunakan bahasa pemrograman C#(CSharp).

3.2.5 Testing

Tahap *testing* adalah tahap pengujian *software* yang telah dibuat. Pengujian ini meliputi fitur-fitur yang diperlukan untuk mengetahui apakah terdapat *error* dan hasilnya sesuai dengan kebutuhan yang sudah didefinisikan di awal.

3.2.6 Maintenance

Tahap ini adalah proses pemeliharaan perangkat lunak. Perangkat lunak yang telah dibuat akan memerlukan pemeliharaan, pemeliharaan ini meliputi penambahan fitur-fitur serta optimasi yang diperlukan nantinya, peningkatan tingkat keamanan dan penyelesaian jika ada *error* yang ditemukan nantinya.

3.3 Peralatan dan Bahan

Peralatan dan bahan yang digunakan dalam perancangan tugas akhir ini meliputi diantaranya:

- a. Laptop.
- b. *Software* Visual Studio 2019.
- c. *Software* Microsoft SQL Server Management Studio 2018.
- d. *Software* Arduino IDE.

e. *Board* Arduino Mega 2560.

3.4 Analisis Kebutuhan Sistem

Kebutuhan sistem secara fungsional yaitu fungsi-fungsi yang harus ada di dalam aplikasi untuk melakukan kerja yang diharapkan. Fungsi-fungsi tersebut adalah:

1. Dapat membangun koneksi melalui *port* sehingga komunikasi serial dengan *board* arduino bisa dilakukan.
2. Pengguna dapat melakukan pemilihan barang-barang.
3. Pengguna dapat menambahkan jumlah pesanan.
4. Sistem dapat mengirimkan data pesanan ke mikrokontroler.
5. Mikrokontroler arduino dapat menerima data pesanan untuk di proses.
6. Aplikasi dapat menerima *feedback* dari mikrokontroler jika pesanan telah selesai di proses.

3.5 Analisis Kebutuhan Perangkat Lunak

3.5.1 Kebutuhan Proses

Beberapa kebutuhan proses dalam pengembangan aplikasi, yaitu:

1. Proses pengaturan *connection port*.
2. Proses pembuatan pesanan.
3. Proses pengiriman data pesanan dan menerima *feedback* dari mikrokontroler.

3.5.2 Kebutuhan Input

Kebutuhan input yang diperlukan, antara lain:

1. Data jumlah barang yang tersedia.
2. Data pesanan.

3.5.3 Kebutuhan Output

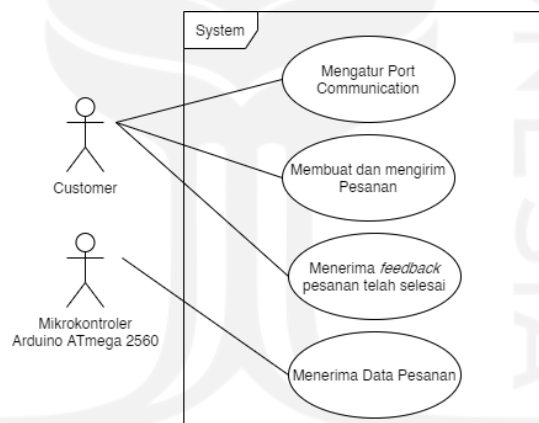
Output yang diharapkan dari aplikasi adalah mampu mengirimkan data pesana ke mikrokontroler arduino untuk diproses ke alat *prototype* dan menerima *feedback* jika proses pesanan telah selesai.

3.6 Perancangan Perangkat Lunak

Metode perancangan perangkat lunak yang digunakan adalah *Unified Modeling Language* (UML). Model UML yang digunakan adalah *Use Case Diagram* dan *Activity Diagram*.

3.6.1 Use Case Diagram

Bentuk *Use Case Diagram* yang digunakan dalam pembuatan aplikasi ini dapat dilihat pada gambar 3.3.



Gambar 3.3: *Use Case Diagram*

Text dibawah ini akan mendeskripsikan *use case diagram* yang ditunjukkan pada gambar 3.3:

1. Terdapat dua aktor yang akan berinteraksi dengan sistem yaitu *customer* dan mikrokontroler arduino ATmega 2560.
2. *Customer* mengatur *port communication* agar aplikasi dapat melakukan komunikasi serial dengan mikrokontroler.
3. *Customer* melakukan pemilihan barang yang diinginkan, membuat pesanan berdasarkan barang yang terpilih kemudian mengirim data pesanan.

4. Mikrokontroler menerima data pesanan dan memprosesnya.
5. *Customer* menerima *feedback* berupa pesan bahwa pesanan telah selesai di proses.

3.6.2 Use Case Diagram Actors Details

1. *Customer* adalah pelanggan yang datang ke toko untuk melakukan pembuatan pesanan melalui aplikasi desktop yang tersedia. Pelanggan melakukan pemilihan barang dan akan melakukan langkah terakhir yaitu proses pesanan. Mengenai data yang dikirimkan oleh aplikasi ke mikrokontroler yang berisi barang-barang yang dipesan dalam satu pesanan akan dijelaskan sebagai berikut.

(a) Bentuk Data yang Dikirim

Gambar 3.4 merupakan paket data yang dikirimkan ke mikrokontroler untuk memproses pesanan di prototipe alat. Paket ini terdiri dari 3 komponen yaitu, simbol tagar/*hashtag*, *placeholder* dan simbol dollar. Simbol tagar/*hashtag* berfungsi sebagai pemisah nilai, *placeholder* seperti 0,1,2 dan 3 berfungsi sebagai tempat sementara yang nantinya akan diganti dengan nilai yang sebenarnya, yaitu jumlah barang yang diinginkan untuk barang 1 sampai barang 4. Terakhir, simbol dollar digunakan sebagai penutup paket data pesanan.

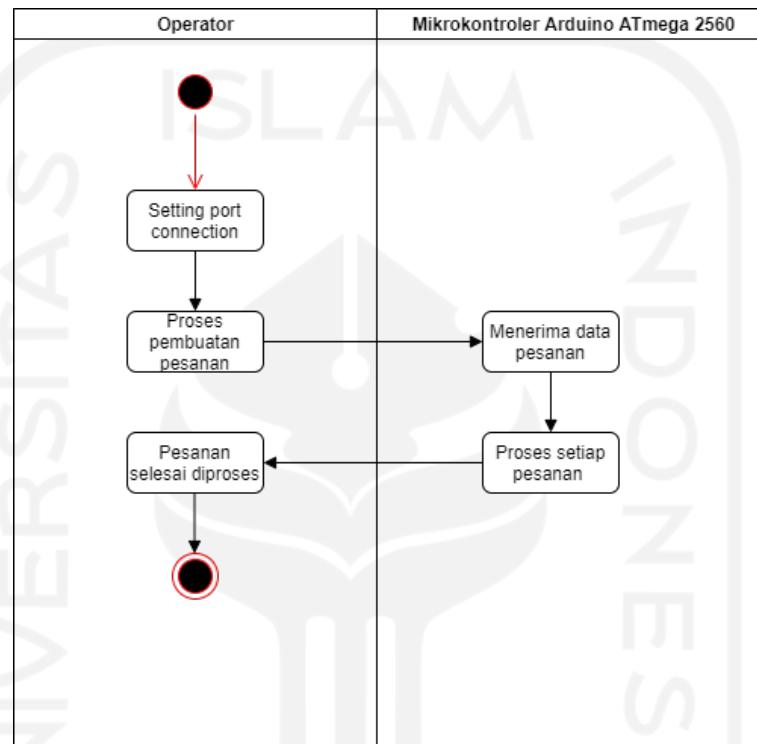
The diagram shows a data packet format: `#{0}#{1}#{2}#{3}#$`. The first four characters are enclosed in a box, and the last character is a dollar sign.

Gambar 3.4: *Paket Data Pesanan*

2. Mikrokontroler Arduino ATmega 2560 adalah kontroler yang berinteraksi dengan sistem dalam hal ini mikrokontroler menerima paket data yang sama seperti pada gambar 3.4. Setelah menerima paket data tersebut yang akan dilakukan oleh mikrokontroler adalah melakukan *parsing* dan mengambil informasi nilai barang 1 sampai barang 4 yang diinginkan, nilai-nilai tersebut akan menggantikan *placeholder*.

3.6.3 Activity Diagram

Berdasarkan *use case* yang telah dibuat, maka dapat dibuat model *activity diagram* untuk menunjukkan aktivitas-aktivitas yang terjadi selama proses program berjalan, *activity diagram* dapat dilihat pada gambar [3.5](#).



Gambar 3.5: Activity diagram

3.7 Rancangan Antarmuka Halaman Setting Port Connection

Rancangan antarmuka halaman *setting port connection* berfungsi untuk menampilkan tampilan ketika operator melakukan pengaturan *port connection*. Rancangan antarmuka halaman ini dapat dilihat pada gambar [3.6](#).

The image shows a 'Settings' dialog box with a white background and a black border. Inside the dialog, there is a smaller rectangular area containing three vertically stacked input fields. The first field is labeled 'Port Name', the second is labeled 'Baudrate', and the third is labeled 'Status'. Below these fields, there are two buttons: 'Connect' on the left and 'Close' on the right. The entire dialog box is centered on the page.

Gambar 3.6: Rancangan Antarmuka Pengaturan *Port Connection*

3.7.1 Pemilihan User Interface Elements Setting Port Connection

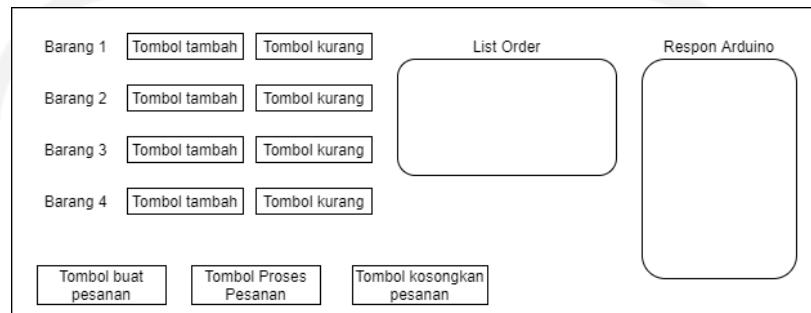
Pemilihan *UI elements* didasarkan pada kebutuhan untuk membangun koneksi sehingga sistem dapat melakukan pertukaran data dengan sistem lain yang terlibat, untuk itu akan dijelaskan sebagai berikut.

Terdapat beberapa *UI elements* yang digunakan, yaitu:

1. *Button* : *button connect* dan *button close* digunakan untuk membangun koneksi dan menutup koneksi dengan *Port* yang tersedia.
2. *Label* : label digunakan untuk memberikan keterangan pada bagian tertentu seperti *Port Name*, *Baudrate*, *Status*.
3. *Combo box* : *combo box* digunakan untuk memberikan berbagai pilihan yang tersedia masing-masing untuk *Port Name*, *Baudrate*.
4. *Progress bar* : *progress bar* digunakan untuk mengindikasikan koneksi berhasil dibangun.

3.8 Rancangan Antarmuka Halaman Proses Pesanan

Rancangan antarmuka halaman proses pesanan berfungsi untuk menampilkan tampilan ketika operator melakukan pembuatan pesanan, mengirim data pesanan ke mikrokontroler dan menerima *feedback* dari mikrokontroler. Rancangan antarmuka halaman ini dapat dilihat pada gambar [3.7](#)



Gambar 3.7: Rancangan Antarmuka Proses Pesanan

3.8.1 Pemilihan User Interface Elements Item Selections

UI elements yang dapat membantu pemilihan barang-barang digunakan pada bagian ini. Terdapat beberapa *UI elements* yang digunakan, yaitu:

1. *Button* : *button* digunakan untuk menambahkan dan mengurangi jumlah barang, juga digunakan untuk membuat pesanan, proses pesanan (mengirimkan data ke mikrokontroler) dan menghapus antrian pesanan di list.
2. *Label* : *label* digunakan untuk memberikan keterangan seperti, barang 1 sampai barang 4, *list order* dan respon arduino.
3. *List view* : *list view* digunakan untuk menampilkan pesanan-pesanan yang telah dibuat dengan menggunakan teknik *data-binding*.
4. *Rich text box* : *rich text box* digunakan untuk menampilkan respon dari arduino berupa kalimat bahwa pesanan telah selesai diproses.

3.9 Metode Pengujian

Pengujian perangkat lunak merupakan aktivitas yang penting untuk dilakukan karena dengan melakukan pengujian maka akan didapatkan informasi mengenai aplikasi yang telah dikembangkan seperti *error* atau *bug* yang terjadi ketika melakukan

aktivitas pada fungsi-fungsi dan yang paling utama adalah memastikan aplikasi dapat berjalan sesuai kebutuhan atau spesifikasi yang diinginkan.

Metode pengujian yang digunakan dalam pengembangan aplikasi ini adalah metode pengujian *black box testing* dan diuji oleh pengembang. *Black box testing* merupakan metode pengujian *software* yang pengujiannya juga bisa dilakukan oleh orang awam yang tidak memiliki pengetahuan teknis. Tujuan dari *black box testing* adalah untuk mengetahui apakah secara fungsional aplikasi dapat menjalankan fungsi-fungsinya sesuai kebutuhan. Fungsi-fungsi yang diujikan harus dapat berjalan sesuai kebutuhan saat awal pendefinisian kebutuhan sistem, kebutuhan-kebutuhan tersebut meliputi:

1. Aplikasi dapat terhubung dengan mikrokontroler sehingga dapat melakukan pengiriman data.
2. Aplikasi memungkinkan operator untuk melakukan pemilihan barang.
3. Aplikasi memungkinkan operator untuk membuat pesanan berisikan barang-barang yang telah dipilih.
4. Aplikasi dapat mengirimkan data pesanan-pesanan ke mikrokontroler untuk diproses.
5. Aplikasi dapat menerima *feedback* dari mikrokontroler jika pesanan telah selesai diproses.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pembangunan Sistem

Pembangunan sistem aplikasi dimulai dari implementasi *User Interface* sesuai desain yang telah dibuat sebelumnya dan fungsi-fungsi yang dibutuhkan agar *system requirement* terpenuhi. Proses implementasi *User Interface* dilakukan dengan cara menggunakan *Extensible Application Markup Language (XAML)* sesuai *WPF Controls* yang dibutuhkan.

Basis data Microsoft SQL Server digunakan untuk menyimpan kuantitas barang atau produk. Jika kuantitas di basis data sama dengan nol untuk barang tertentu, maka saat pemilihan produk, produk tersebut tidak dapat ditambahkan.

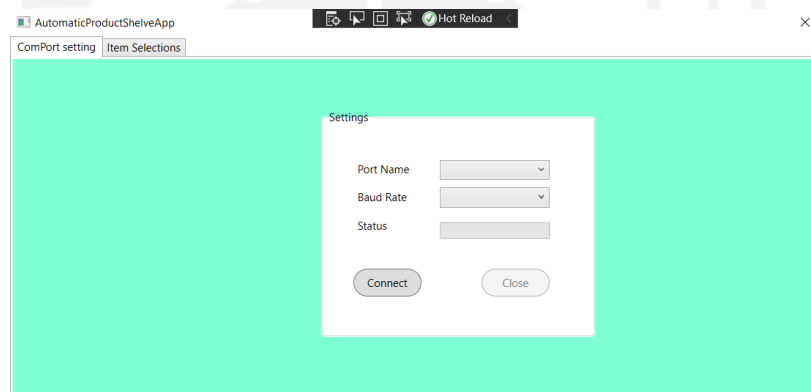
Mikrokontroler arduino ATmega 2560 akan menerima data yang dikirim dari aplikasi C# kemudian menjalankan program atau instruksi-instruksi yang telah dibuat hingga memproses pesanan satu per satu, kemudian setelah setiap pesanan selesai diproses maka akan mengirimkan *feedback* atau respon kembali ke aplikasi C# untuk menginformasikan bahwa pesanan telah selesai di proses.

4.2 Implementasi Pengujian User Interface

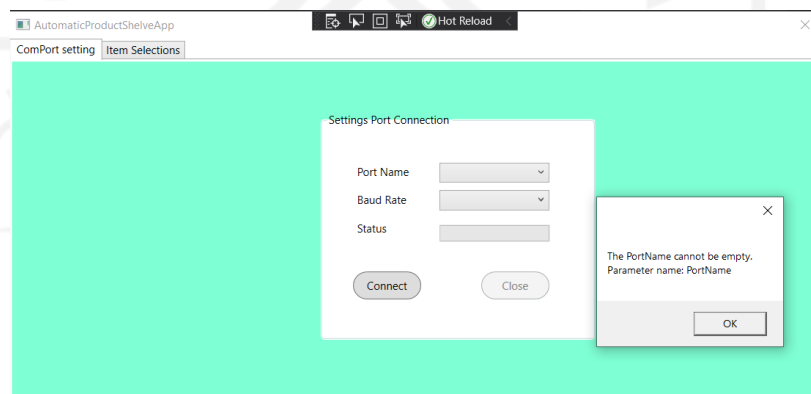
Implementasi dilakukan menggunakan teknologi WPF yang terkait bahasa pemrograman XAML sebagai pendukung bahasa pemrograman C# dalam pengembangan aplikasi dekstop.

4.2.1 Halaman Communication Port

Halaman *Communication Port* merupakan halaman untuk membangun koneksi dengan sistem luar yang akan berinteraksi dengan aplikasi C# sehingga dapat terjadi pertukaran data. Halaman ini berfungsi untuk memastikan dan mengatur bahwa *port communication* seperti COM4 atau nama yang lain dapat terlihat, dipilih dan terhubung dengan baik. *Baud rate* yang merupakan kecepatan transfer data dalam satuan BPS atau Bits per Second juga dapat diatur sesuai kebutuhan agar proses pengiriman data tidak mengalami *lagging*. *Button* untuk membangun dan menutup koneksi juga tersedia sehingga untuk membangun koneksi atau menutup diperlukan *trigger* dari *button-button* ini. Jika *button open* tidak digunakan maka koneksi tidak akan terbangun dan pertukaran data dengan sistem luar tidak dapat dilakukan.



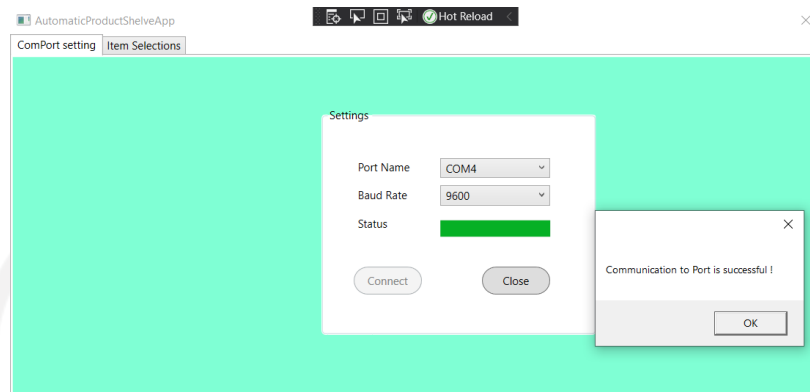
Gambar 4.1: Halaman *Communication Port*



Gambar 4.2: Dialog box untuk *port name* kosong

Ketika *button "connect"* ditekan namun nama port belum dipilih maka dialog box seperti pada Gambar 4.2 yang menyatakan port name tidak boleh kosong akan

keluar dan koneksi tidak dapat terbangun.



Gambar 4.3: Dialog box muncul mengindikasikan koneksi berhasil terbangun

Koneksi port yang berhasil terbangun diindikasikan dengan status bar terisi penuh berwarna hijau dan muncul dialog box menyampaika pesan "*Communication to port is successful!*", jika dialog box tersebut muncul maka sistem dapat melakukan proses pertukaran data dengan sistem luar dalam hal ini adalah mikrokontroler arduino ATmega 2560.

4.2.2 Penggunaan Controls WPF Communication Port

Controls WPF Communication Port		
Controls	Posisi	Fungsi
Canvas Panel	Center	Positioning child elements
Group Box	Center	Grouping controls yang berhubungan
Label	Port Name, Baud Rate, Status	Menuliskan text atau bisa sebagai access key
Combo Box	Port Name dan Baud Rate	Memunculkan items drop-down list agar dapat dipilih oleh user
Progress Bar	Status bar	Mengindikasikan progress koneksi melalui bar
Button	Button Connect	Membangun koneksi sesuai user input
Button	Button Close	Menutup koneksi

Tabel 4.1: WPF Controls Halaman Communication Port

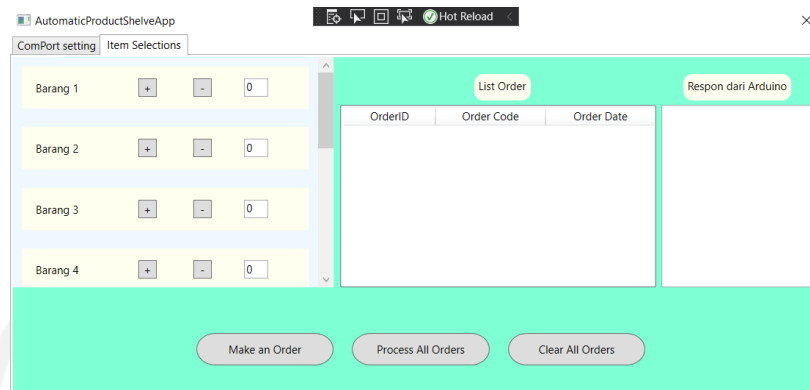
Penggunaan controls WPF seperti pada tabel 4.1 digunakan untuk membantu pengembangan perangkat lunak berbasis desktop sesuai kebutuhan yang diperlukan dalam membangun koneksi agar sistem dapat melakukan pertukaran data antar sistem melalui komunikasi serial.

4.2.3 Halaman Items Selection

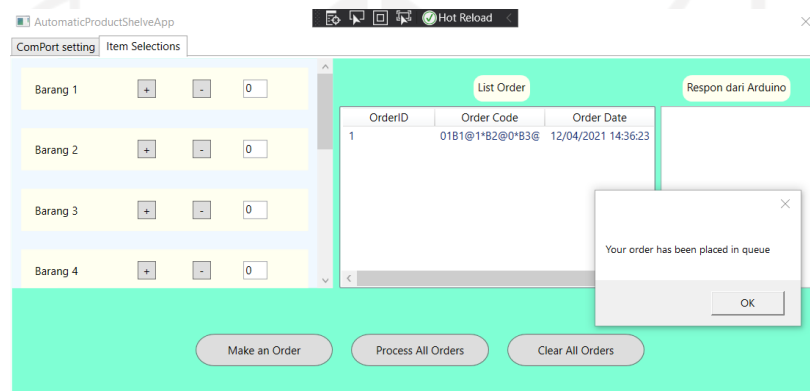
Setelah membangun koneksi untuk melakukan pertukaran data di halaman *communication port*, maka halaman *Items Selection* digunakan untuk melakukan pemilihan barang atau produk-produk yang tersedia beserta kuantitas yang diinginkan, membuat pesanan dengan tombol "Make an order". Setelah itu pesanan yang dibuat akan masuk kedalam antrian "list pesanan" dan semua pesanan yang telah dibuat dan masuk kedalam "list pesanan" dapat diproses menggunakan tombol "Process All Orders". Tombol "Clear All Orders" digunakan untuk menghapus seluruh pesanan yang ada di dalam antrian.

Jika koneksi pada port tidak terbangun maka tombol untuk melakukan pembuatan pesanan tidak dapat berfungsi, sama halnya dengan tombol yang lain untuk melakukan proses pesanan.

Pembuatan pesanan ditunjukkan pada Gambar 4.5 melalui tombol "Make an Order" berhasil dilakukan. Pesanan tersebut kemudian ditambahkan kedalam antrian

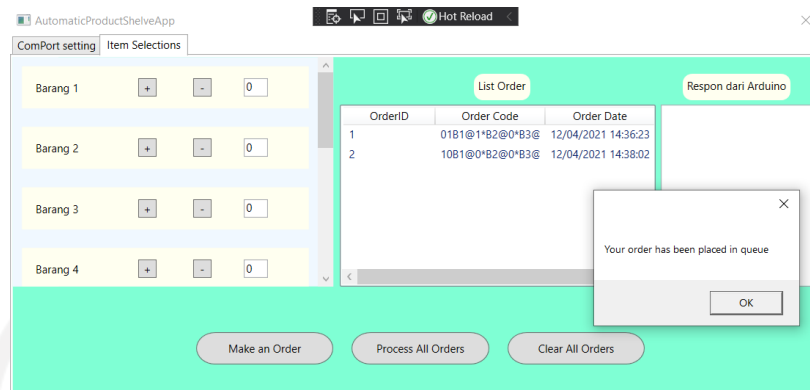


Gambar 4.4: Halaman *Items Selection*



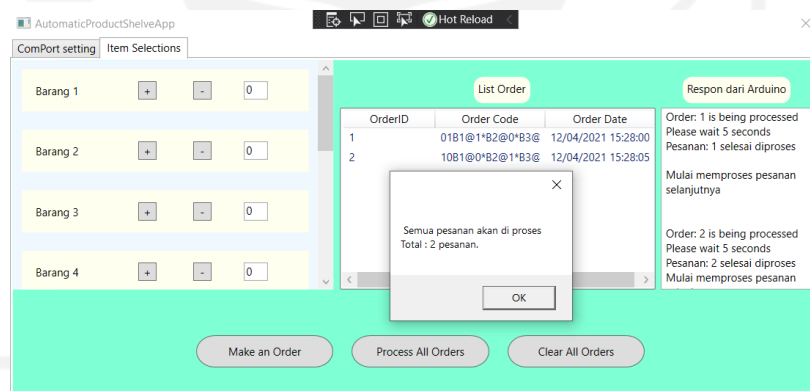
Gambar 4.5: Satu pesanan berhasil ditambahkan ke dalam list antrian

dan terlihat di bagian "List Order", data-data yang muncul adalah *OrderID*, *order code* dan *order data*. Dialog Box muncul mengindikasikan pesanan berhasil diletakkan di dalam antrian.



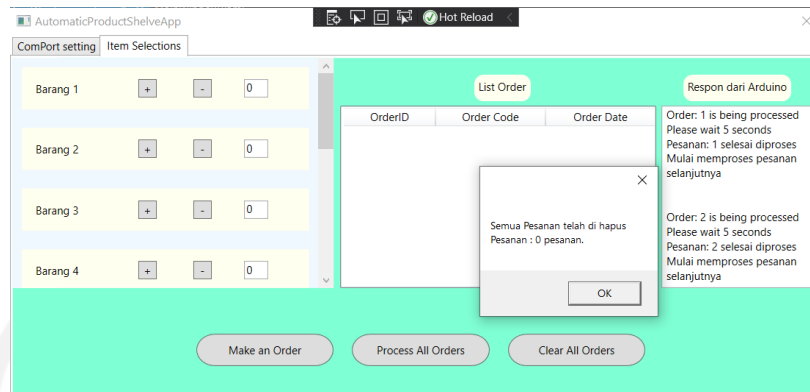
Gambar 4.6: Pesanan lain berhasil ditambahkan ke dalam list antrian

Penambahan pesanan yang dibuat ditunjukkan pada Gambar 4.6 melalui "List Order" dimana pesanan kedua memiliki OrderID 2 dan Order Code serta Order Date yang berbeda dengan pesanan yang sebelumnya. Setiap pesanan yang berhasil ditambahkan kedalam antrian, dialog box akan muncul mengindikasikan penambahan berhasil.



Gambar 4.7: Proses setiap pesanan

Proses setiap pesanan artinya mengirimkan data kepada mikrokontroler arduino ATmega 2560, menyelesaikan proses di purwarupa kemudian mengirimkan *feedback* kembali ke aplikasi C# bahwa pesanan telah selesai yang ditampilkan pada *control* "rich text box" dan akan memproses pesanan selanjutnya seperti yang ditunjukkan pada Gambar 4.7. Proses ini dicapai dengan menggunakan tombol "Process All Orders".



Gambar 4.8: Menghapus seluruh pesanan di antrian

Penghapusan seluruh pesanan ditunjukkan pada Gambar 4.8 artinya menghapus seluruh pesanan yang sebelumnya telah ada di dalam antrian. Hal ini dicapai dengan menekan tombol "*Clear All Orders*". Setelah menghapus seluruh pesanan dan ingin membuat pesanan lagi, maka pesanan akan dimulai dari nomor urut satu kembali.

4.2.4 Penggunaan Controls WPF Items Selection

Controls WPF Items Selection		
Controls	Posisi	Fungsi
Canvas Panel	Center	Positioning Child Elements
Scroll Viewer	Pemilihan Produk	Container agar konten didalamnya bisa dilihat menggunakan scrolling
Stack Panel	Pemilihan Produk	Membuat child elements bisa dibuat horizontal atau vertical orientasinya
Text Box	Kuantitas Produk	Digunakan untuk text-input control
Label	List Order dan Respon Arduino	Menuliskan text atau bisa sebagai access key
Button	Button "Make an Order"	Menambahkan pesanan ke list

Tabel 4.2: WPF Controls Halaman Items Selection (bagian 1)

Pada Tabel 4.2 ditunjukkan *control-control* WPF yang digunakan dalam pengembangan aplikasi dekstop untuk halaman *items selection*. Bagian ini dibuat menjadi 2 bagian untuk memudahkan penglihatan dalam melihat tabel.

Controls WPF Items Selection		
Controls	Posisi	Fungsi
Button	Button Process All Orders	Mengirimkan setiap data pesanan ke mikrokontroler
Button	Button Clear All Orders	Menghapus seluruh pesanan dalam antrian
Rich Text Box	Respon Arduino	Menampilkan respon yang diterima dari mikrokontroler
List View	List Order	Dengan menggunakan data-binding dapat menampilkan data-data yang ada di class (i.e. data pesanan-pesanan)
Tab Item	Center	Digunakan untuk split area yang berbeda dengan navigasi tab control

Tabel 4.3: WPF Controls Halaman Items Selection (bagian 2)

Pada Table 4.3 ditunjukkan *control-control* WPF lain yang digunakan dalam pembuatan halaman *items selection*. *Control-control* tersebut mampu memenuhi kebutuhan yang diinginkan agar sistem berjalan sesuai keinginan dalam hal memproses pesanan.

4.3 Implementasi Pengujian Sistem

4.3.1 Pengujian User Interface

Pengujian *user interface* digunakan untuk mengetahui apakah UI elemen yang digunakan dapat berfungsi sesuai yang diinginkan. *User Interface* (UI) elemen tersebut meliputi tombol-tombol yang merespon dari input yang diberikan oleh *user*. Hasil pengujian *user interface* dapat dilihat pada tabel 4.4 dan 4.5 dibawah ini.

No	Kasus Diuji	Hasil Diinginkan	Hasil Didapat	Status
1	Tombol "Connect" di halaman Communication Port	Sistem dapat membangun koneksi dengan port yang dituju	Setelah tombol "Connect" diklik, dialog box muncul, status bar full dan koneksi port dapat terhubung	Berhasil
2	Tombol "Close" di halaman Communication Port	Sistem dapat menutup koneksi port	Setelah tombol "Close" diklik, status bar akan menjadi nol dan koneksi port tertutup.	Berhasil
3	Tombol "Tambah" di halaman Items Selection untuk setiap produk	Sistem dapat menambahkan kuantitas setiap produk yang diinginkan	Setelah tombol diklik maka kuantitas produk meningkat 1 , ditunjukkan pada text box	Berhasil

Tabel 4.4: Hasil Pengujian User Interface (bagian 1)

No	Kasus Diuji	Hasil Diinginkan	Hasil Didapat	Status
4	Tombol "Kurang" di halaman Items Selection untuk setiap produk	Sistem dapat mengurangi kuantitas setiap produk yang diinginkan	Setelah tombol diklik maka kuantitas produk berkurang 1, ditunjukkan pada text box	Berhasil
5	Tombol "Make an Order" di halaman Items Selection	Sistem dapat mengambil data terkait produk yang dipilih dan membuatnya menjadi pesanan	Setelah tombol diklik maka pesanan akan terbuat dan ditampilkan pada "List Order"	Berhasil
6	Tombol "Process All Order" di halaman Items Selection	Sistem dapat mengirimkan setiap data pesanan ke mikrokontroler untuk diproses	Setelah tombol diklik maka data pesanan-pesanan dikirimkan ke mikrokontroler untuk diproses dan dialog box muncul menyatakan jumlah pesanan yang di proses	Berhasil
7	Tombol "Clear All Order" di halaman Items Selection	Sistem dapat menghapus pesanan-pesanan yang ada di dalam antrian atau "List order"	Setelah tombol diklik maka pesanan-pesanan yang ada di dalam antrian terhapus dan dialog box akan muncul	Berhasil

Tabel 4.5: Hasil Pengujian User Interface (bagian 2)

4.3.2 Pengujian Dasar Sistem

Pengujian dasar sistem merupakan uji fungsi-fungsi dasar sistem sesuai kebutuhan untuk melakukan pembuatan pesanan dan memproses pesanan dengan keterlibatan mikrokontroler arduino ATmega 2560. Hasil Pengujian dasar sistem bisa dilihat pada tabel 4.6 dan tabel 4.7 di bawah ini.

No	Fungsi Diuji	Hasil Diharapkan	Hasil Didapat	Status
1	Fungsi untuk membangun koneksi agar sistem dapat berinteraksi dengan sistem luar	Sistem dapat membangun koneksi melalui COM atau nama lain yang terdeteksi yaitu sistem luar	Sistem berhasil membangun koneksi agar komunikasi serial dapat terjadi	Berhasil
2	Fungsi untuk melakukan pemilihan produk-produk	Sistem memberikan pilihan produk-produk yang dapat dipilih	Sistem berhasil menyediakan bagian untuk melakukan pemilihan produk	Berhasil
3	Fungsi untuk menambahkan dan mengurangi kuantitas produk yang dipilih yang berkaitan dengan data di basis data	Sistem mampu menambahkan kuantitas produk yang dipilih jika stok produk di basis data masih tersedia serta mengurangkannya.	Sistem berhasil untuk mengakses basis data guna dalam penambahan dan pengurangan produk yang dipilih	Berhasil

Tabel 4.6: Hasil Pengujian Fungsi (bagian 1)

No	Fungsi Diuji	Hasil Diharapkan	Hasil Didapat	Status
4	Fungsi untuk menampilkan pesanan yang telah dibuat	Sistem mampu menampilkan pesanan-pesanan yang telah dibuat.	Sistem berhasil menampilkan pesanan-pesanan yang dibuat dan masuk kedalam antrian	Berhasil
5	Fungsi untuk memproses antrian pesanan	Sistem mampu mengirimkan data pesanan-pesanan ke mikrokontroler arduino ATmega 2560	Sistem berhasil mengirimkan data pesanan-pesanan ke mikrokontroler arduino ATmega 2560	Berhasil
6	Fungsi untuk menerima feedback dari mikrokontroler Arduino ATmega 2560	Sistem mampu menampilkan feedback atau respon dari mikrokontroler arduino ATmega 2560 ketika selesai memproses pesanan	Sistem berhasil menerima dan menampilkan feedback dari mikrokontroler arduino ATmega 2560	Berhasil

Tabel 4.7: Hasil Pengujian Fungsi (bagian 2)

4.4 Hasil Prototipe Alat

Hasil dari prototipe alat dapat dilihat pada gambar [4.9](#), [4.10](#), [4.11](#). Prototipe alat terdiri dari beberapa komponen utama seperti *box ejector* untuk mendorong kotak, motor stepper untuk menggerakkan konveyor, motor dc untuk menjatuhkan barang (untuk barang 1, 2, 3 dan 4), sensor induksi untuk mengecek kotak yang melewati konveyor dan sensor objek sebagai *counter* setiap barang yang jatuh.



Gambar 4.9: Prototipe Tampak Depan



Gambar 4.10: Prototipe Tampak Samping



Gambar 4.11: Prototipe Tampak Belakang

4.5 Hasil Pengujian Perangkat lunak dengan Prototipe Alat

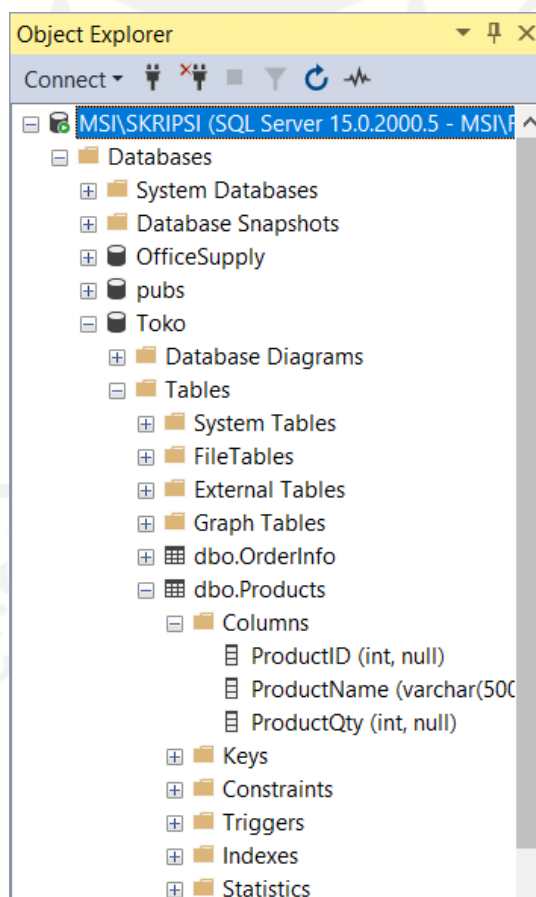
Pada tabel 4.8 ditunjukkan hasil pengujian prototipe alat untuk berbagai pesanan / *order*. Terdapat 15 percobaan dan meliputi kombinasi yang berbeda karena secara prinsip prototipe mensimulasikan kerja alat untuk 4 jenis barang yang berbeda yaitu A,B,C dan D. Setelah 4 jenis barang berbeda diuji coba melalui pesanan, kemudian diuji menggunakan kombinasi lain dengan 2 jenis barang, 3 jenis barang dan 4 jenis barang. Hasilnya, pada setiap percobaan untuk berbagai kombinasi pesanan/*order* dikategorikan berhasil karena secara fungsional perangkat lunak mampu memenuhi kebutuhan dan sistem tidak mengalami error saat dijalankan.

Terkait adanya perbedaan waktu yang terjadi pada percobaan satu sampai tiga untuk setiap kombinasi disebabkan karena ada beberapa kendala yang mempengaruhi, yaitu posisi box berubah sehingga tidak masuk *drop zone*, barang tersangkut di rak, box tidak terbaca sensor, barang tersangkut di bibir box.

No	Order	Hasil						Rata-rata waktu (detik)	Tingkat Keberhasilan
		Percobaan 1		Percobaan 2		Percobaan 3			
		Keterangan	Waktu (detik)	Keterangan	Waktu (detik)	Keterangan	Waktu (detik)		
1	A	Berhasil	11	Berhasil	12	Berhasil	18	13,67	3/3
2	B	Berhasil	11	Berhasil	12	Berhasil	14	12,33	3/3
3	C	Berhasil	11,84	Berhasil	13,09	Berhasil	11,95	12,29	3/3
4	D	Berhasil	17,89	Berhasil	12,04	Berhasil	12,2	14,04	3/3
5	AB	Berhasil	16	Berhasil	15	Berhasil	16,05	15,68	3/3
6	AC	Berhasil	16	Berhasil	19,87	Berhasil	23,27	19,71	3/3
7	AD	Berhasil	14,13	Berhasil	14,08	Berhasil	21,11	16,44	3/3
8	BC	Berhasil	15,08	Berhasil	13,25	Berhasil	14,08	14,14	3/3
9	BD	Berhasil	16,27	Berhasil	13,12	Berhasil	24,15	17,85	3/3
10	CD	Berhasil	14,11	Berhasil	21,21	Berhasil	14,2	16,51	3/3
11	ABC	Berhasil	17,04	Berhasil	13,02	Berhasil	16,22	15,43	3/3
12	ABD	Berhasil	17,08	Berhasil	17,11	Berhasil	21,14	18,44	3/3
13	ACD	Berhasil	18,2	Berhasil	17,04	Berhasil	17	17,41	3/3
14	BCD	Berhasil	17,03	Berhasil	19,2	Berhasil	16,24	17,49	3/3
15	ABCD	Berhasil	24,21	Berhasil	20,16	Berhasil	19,21	21,19	3/3

Tabel 4.8: Hasil Pengujian Prototipe Alat

4.6 Basis Data



Gambar 4.12: Tabel dbo.Products di basis data

Tabel `dbo.Products` pada gambar 4.9 terdiri dari tiga kolom yaitu `ProductID` bertipe `int`, `ProductName` bertipe `varchar` dan `ProductQty` bertipe `int`. Tabel ini menyimpan data produk-produk yang tersedia untuk purwarupa alat.

```
private void TambahBarang1()
{
    // connection string ke database.
    string connectionString;
    SqlConnection cnn;
    connectionString = @"Data Source=MSI\SKRIPSI;Initial Catalog=Toko;Integrated Security=true";
    cnn = new SqlConnection(connectionString);
    cnn.Open();

    SqlCommand command;
    SqlDataReader dataReader;
    String sql, output = "";

    sql = "select ProductQty from Products where ProductName = 'Barang 1'";
    command = new SqlCommand(sql, cnn);
    dataReader = command.ExecuteReader();

    // cek apakah kuantitas barang 1 ada dan lebih dari 0.
    if (dataReader.Read())
    {
        output += dataReader.GetValue(0);
        if (int.Parse(output) > 0)
        {
            result1 += 1;
            txtBarang1.Text = result1.ToString();
            UpdateDatabaseMin.Barang1();
            dataReader.Close();
            cnn.Close();
        }
    }
    //UpdateBarang();
}
```

Gambar 4.13: Fungsi akses dan update data di basis data (1)

Pada gambar 4.10 ditunjukkan bagaimana mengambil data dari basis data ketika tombol tambah ditekan. Penambahan dapat dilakukan selama kuantitas produk di basis data masih tersedia untuk produk yang dipilih, jika kuantitas atau stok nol maka penambahan tidak dapat dilakukan.

```
        result1 -= 1;
        txtBarang1.Text = result1.ToString();
        UpdateDatabaseMin.Barang1();
        dataReader.Close();
        cnn.Close();
    }
    //UpdateBarang();
}
}
1 reference
private void KurangBarang1()
{
    if (result1 > 0)
    {
        result1 -= 1;
        txtBarang1.Text = result1.ToString();
        UpdateDatabasePlus.Barang1();
    }
    //UpdateBarang();
}
```

Gambar 4.14: Fungsi akses dan update data di basis data (2)

Pada gambar 4.11 ditunjukkan bagaimana fungsi untuk tombol kurang sehingga barang yang tidak jadi dipilih atau dikurangkan akan kembali *update* basis data sesuai produk yang dikurangkan.

4.7 Analisis dan Pembahasan

4.7.1 Proses Pertukaran Data

Proses pertukaran data dilakukan dengan menggunakan komunikasi serial, komunikasi serial merupakan tipe komunikasi dimana 1 bit data dikirimkan dalam satu waktu secara berurutan melalui komunikasi bus. Data pada gambar 3.4 dikirim dari *Least Significant Bit* (LSB) sampai ke *Most Significant Bit* (MSB) secara berurutan antar sistem.

4.7.2 Synchronous Serial Protocol

Serial protokol yang digunakan adalah *synchronous* menggunakan *Universal Serial bus* (USB). USB memudahkan transmisi data karena penggunaannya yang mudah yaitu sebagai "*plug and play*" device.

4.7.3 Siklus Pemecahan Masalah

Pendekatan pemecahan masalah / *problem solving* yang saya gunakan dalam banyak kasus adalah "*Divide and Conquer*". Pendekatan ini berfokus pada suatu masalah yang cukup besar kemudian membaginya (*divide*) menjadi bagian-bagian kecil sampai menjadi mudah kemudian menyelesaikan (*conquer*) satu per satu masalah kecil tadi hingga solusi untuk keseluruhan masalah awal dapat dihasilkan seperti dalam konteks penulisan program sesuai yang diinginkan. Pendekatan ini dapat digunakan di hampir semua kondisi masalah.

4.7.4 Analisis User Interface

User Interface yang dibuat dengan bantuan *controls* WPF dapat berjalan dengan baik sesuai yang diharapkan. Elemen-elemen UI tersebut dapat merespon input dari user sehingga ketika tombol-tombol ditekan sesuai keadaan yang dibutuhkan, maka proses-proses yang diharapkan dapat terjadi. Elemen-elemen UI tersebut ditunjukkan pada gambar 4.1, gambar 4.2 dan gambar 4.3 untuk halaman *Communication Port*, serta pada tabel 4.1 ditunjukkan *controls* WPF yang digunakan. Elemen-elemen UI untuk halaman *Items Selection* ditunjukkan pada gambar 4.4 serta *controls* WPF yang digunakan ditunjukkan pada tabel 4.2 dan 4.3.

Tampilan yang sudah dibuat masi sangat sederhana belum menerapkan prinsip-prinsip UI/UX yang baik dan benar walaupun secara fungsional dapat berjalan sesuai yang

diinginkan, oleh karena itu terdapat banyak ruang untuk perbaikan dan penambahan fitur.

4.7.5 Mikrokontroler Proses Pesanan Algoritma

Dalam memproses pesanan yang masuk, mikrokontroler bekerja sesuai algoritma yang telah dibuat, algoritma pada program mikrokontroler dapat dijelaskan sebagai berikut.

1. Cek apakah ada data pesanan yang masuk.
2. Jika tidak ada, semua motor off.
3. Jika ada, baca kode pesanan.
4. Jika kode pesanan salah, kembali baca kode pesanan.
5. Jika kode pesanan benar, lakukan *parsing* dan ambil informasi jumlah barang 1 sampai barang 4.
6. Dorong *reload box* keluar.
7. Putar motor stepper (konveyor).
8. Proses barang di rak 1 (2 motor dc).
9. Jika Proses di rak 1 selesai, proses barang di rak 2 (2 motor dc).
10. Jika proses di rak 2 selesai, motor stepper (konveyor) off.
11. Kembali ke langkah 1.

4.7.6 Analisis Pengujian Sistem

Pengujian sistem seperti yang ditunjukkan pada tabel 4.6 dan tabel 4.7 berhasil dilakukan. Sistem mampu melakukan pembangunan koneksi dengan sistem luar (mikrokontroler arduino ATmega 2560) untuk melakukan komunikasi serial, sistem mampu melakukan pembuatan pesanan sesuai produk-produk yang dipilih, sistem mampu menampilkan pesanan yang dibuat dan mengirimkan data-data pesanan ke mikrokontroler arduino ATmega 2560 untuk diproses di purwarupa alat dan sistem mampu menampilkan *feedback* atau respon dari mikrokontroler arduino ATmega 2560 ketika selesai memproses tiap pesanan.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis dan pengujian fungsional aplikasi ini, didapat kesimpulan sebagai berikut:

1. Aplikasi dapat membangun koneksi port untuk melakukan komunikasi serial dengan mikrokontroler Arduino ATmega 2560.
2. Aplikasi dapat melakukan pemilihan produk pada halaman *Items Selection*.
3. Aplikasi dapat mengirim paket data pesanan ke mikrokontroler arduino ATmega 2560.
4. Aplikasi dapat menerima feedback dari mikrokontroler arduino ATmega 2560 setelah selesai memproses setiap pesanan.

5.2 Saran

1. Mengembangkan tampilan agar lebih baik lagi sesuai kebutuhan namun tetap mudah digunakan seperti penambahan gambar setiap produk yang tersedia.
2. Penelitian selanjutnya dapat membuat algoritma pada mikrokontroler arduino agar proses pesanan dapat dikerjakan dalam satu waktu agar efisien.
3. Pada penelitian selanjutnya diharapkan dapat membuat kode program C# lebih efisien.

DAFTAR PUSTAKA

- [Badiozamany, 2010] Badiozamany, S. (2010). Microsoft SQL Server OLAP Solution – A Survey. (September).
- [Balaji, 2012] Balaji, S. (2012). Waterfall vs v-model vs agile : A comparative study on SDLC. *WATEERFALL Vs V-MODEL Vs AGILE : A COMPARATIVE STUDY ON SDLC*, 2(1):26–30.
- [Cheng, 2019] Cheng, A. (2019). Why Amazon Go May Soon Change The Way We Shop.
- [Clark, 2011] Clark, D. (2011). *Beginning C# Object-Oriented Programming*. Apress, 1 edition.
- [Edwan et al., 2020] Edwan, T. A., Tahat, A., Hammouri, S., Hashem, L., and Da'boul, L. (2020). An intelligent and automated approach for smart minimarkets. *International Journal of Computational Intelligence Systems*, 13(1):852–863.
- [Julien Bayle, 2013] Julien Bayle (2013). *C Programming for Arduino*.
- [Karhe and Patil, 2015] Karhe, R. R. and Patil, A. J. (2015). 3D Model Design using Arduino. *International Journal of Advanced Engineering Research and Science (IJAERS)*, 2(12):49–51.
- [Microsoft, 2019] Microsoft (2019). Welcome to the Visual Studio IDE.
- [Microsoft, 2020] Microsoft (2020). Download SQL Server Management Studio (SSMS) - SQL Server Management Studio (SSMS).
- [Nakov and &, 2013] Nakov, S. and &, C. (2013). *Fundamentals of Computer Programming with C#*. Sofia.
- [Nathan, 2010] Nathan, A. (2010). *WPF 4 UNLEASHED*. Sams, Indianapolis, 1st editio edition.
- [Spotless, 2017] Spotless (2017). Analysing Argos' In-Store Digital User Experience Upgrade | Spotless.