



# **Implementation of Enterprise Resource Planning (ERP) Software**

Final Year Graduation Project  
Submitted to the Department of Computers & Systems Engineering  
in Partial Fulfillment of the Requirements for the Degree of

**Bachelor of Science (B.Sc.) in Electrical Engineering  
(Computers and Systems)**

**Mohamed Elsayed Kamal  
Mohamed Adel  
Rana Ahmed  
Menna Essam  
Abdullah Hesham  
Mohamed Gamal  
Abdelrahman Mohamed**

**Supervised By**

**Dr. Haitham Abo Bakr  
Dr. Ahmed Alenany**

# Declaration

We hereby certify that this project is entirely of our own work and, to the best of our knowledge, does not breach any law of copyright. The use of ideas taken from other sources has been appropriately cited and acknowledged within the text.

# Abstract

**Context.** Nowadays, there is a continuous increase in the number of businesses, firms, and companies. A company or a firm comprises several departments such as manufacturing, sales, marketing, logistics, and accounting. The overall performance of a business, crucially, depends on how well such processes are interacting. As a company or a business grows, the needs of the company also grow, and handling different systems and processes gets more complex. Therefore, there is an urgent need for the automation of the businesses in order to achieve their goals under the supervision of the users. The ultimate goal is to ensure products are well made and delivered to customers in time. Unfortunately, the available solutions are prohibitively expensive and cannot be afforded by start-up businesses or already existing small or medium-sized businesses.

This project attempts to solve this problem by integrating two solutions for business processes: Enterprise Resource Planning (ERP) and Business Process Management (BPM). An ERP software system supports the execution of business processes by integrating tasks related to sales, marketing, manufacturing, logistics, and accounting throughout a business. In addition to this cross-functional integration, companies connect their ERP systems to coordinate business processes with their customers and suppliers. On the other hand, BPM focuses on capturing and improving business processes to make an organization more efficient. This is achieved by first capturing organizations' current state end-to-end processes and then documenting the steps in process maps. While the ERP job cares more about cross-functional integration and tends to be limited to organizational functions, BPM job tends to be much more process-focused.

# Contents

<b>Declaration</b>	<b>1</b>
<b>Abstract</b>	<b>2</b>
<b>I ERP</b>	<b>8</b>
<b>1 Introduction</b>	<b>9</b>
<b>2 ERP</b>	<b>10</b>
<b>3 The ERP</b>	<b>12</b>
I Main System . . . . .	12
(1) System Features . . . . .	12
(2) Development . . . . .	12
(3) Tools . . . . .	14
(4) System Structure . . . . .	14
II The Modules . . . . .	18
(1) CRM . . . . .	19
(2) Warehouse management . . . . .	21
(3) Accounting . . . . .	24
(4) Manufacturing . . . . .	27
(5) Product Manager . . . . .	30
<b>4 System Access</b>	<b>31</b>
I The ERP . . . . .	31
II Mobile Application . . . . .	31
(1) ERP App Main Task . . . . .	31
(2) Main Features . . . . .	31
(3) Application Screens . . . . .	32
<b>II BPM</b>	<b>33</b>
<b>5 Introduction</b>	<b>34</b>
I What is a process ? . . . . .	36
II What Is Not a Process ? . . . . .	36
<b>6 The BPM</b>	<b>37</b>
I The Process . . . . .	37

	(1)      Involved Process Roles . . . . .	38
II	Notation . . . . .	39
	(1)      The Nodes . . . . .	39
III	Tasks . . . . .	42
IV	Variables . . . . .	43
V	The BPM Cycle . . . . .	43
	(1)      Designer . . . . .	44
	(2)      Engine . . . . .	45
	(3)      Monitor . . . . .	46
<b>7</b>	<b>BPM Examples</b>	<b>47</b>
I	Examples of Process-Based Applications . . . . .	47
	(1)      Bakery . . . . .	47
	(2)      Payment Request . . . . .	48
<b>III</b>	<b>Hardware</b>	<b>49</b>
<b>8</b>	<b>The Factory</b>	<b>50</b>
<b>9</b>	<b>Hardware</b>	<b>52</b>
I	Factory box Design . . . . .	52
	(1)      The sorting part . . . . .	53
	(2)      Warehousing part . . . . .	53
	(3)      Collecting part . . . . .	53
	(4)      The Base . . . . .	54
II	components used . . . . .	54
	(1)      TCS230 RGB Color Sensor . . . . .	55
	(2)      Servo motor . . . . .	56
	(3)      Arduino Uno . . . . .	56
	(4)      Node MCU . . . . .	57
<b>10</b>	<b>Software</b>	<b>59</b>
I	Server Side . . . . .	59
II	Client Side . . . . .	61
III	Client App . . . . .	62
	(1)      QR Code . . . . .	63
	(2)      Process Flow . . . . .	64
IV	Important screens . . . . .	64
V	Driver App . . . . .	64
	(1)      Process Flow . . . . .	65
	(2)      Important screens . . . . .	66

<b>IV</b>	<b>Integration</b>	<b>68</b>
<b>11</b>	<b>The Factory Integration</b>	<b>69</b>
I	Manufacturing . . . . .	69
II	Online Market . . . . .	69
III	Warehousing . . . . .	69
IV	Delivering . . . . .	70
V	Database . . . . .	70
VI	Executing Tasks . . . . .	70
<b>12</b>	<b>ERP - BPM - Factory</b>	<b>71</b>
	<b>References</b>	<b>72</b>

# List of Figures

3.1	ERP System Structure . . . . .	15
3.2	Execution Pipe . . . . .	17
5.1	BPM Cycle . . . . .	35
6.1	User Task . . . . .	42
6.2	Tasks Worker . . . . .	43
6.3	Designer . . . . .	44
7.1	Bakery Worflow . . . . .	47
7.2	Bakery Worflow . . . . .	47
7.3	Payment Request Worflow . . . . .	48
7.4	Bakery Worflow . . . . .	48
8.1	The Factory Flow . . . . .	51
9.1	Factory Box . . . . .	52
9.2	Sorting Part . . . . .	53
9.3	Warehousing Part . . . . .	54
9.4	Collecting Part . . . . .	54
9.5	The Base . . . . .	54
9.6	Sorting Connection diagram . . . . .	57
9.7	Collecting Connection diagram . . . . .	58
10.1	Process Flow of the Client Application . . . . .	63
10.2	Making Order Flow . . . . .	65
10.3	Process Flow of the Driver Application . . . . .	66
10.4	Finishing Order Flow . . . . .	67

# Abbreviations

ERP	Enterprise Resource Planning
ORM	Object Relationship Mangment
DLL	Dynamic Link Library
BPM	Business Process Mangment
CRM	Custtomer Relationship Mangment
WMS	Warehouse Mangment System
The ERP	Our implementation of an ERP system
The BPM	Our implementation of an BPM system

# **Part I**

## **ERP**

# Chapter 1

## Introduction

Every business has its own objectives, processes, and requirements. Above all, today's businesses need technologies with complete functions which can bridge the gap between business processes and people. To run a large organization with multiple departments and teams successfully, an ERP system gives a helping hand by synchronizing all information and communication within the organization. ERP is a combination of software and company's activities performed to manage operations. With ERP software, the entire project value chain is aligned and critical processes are streamlined effectively.

Under an unmanaged system, various business processes within an organization utilize multiple disparate applications to manage similar operations. This leads to chaotic data transfer, time-consuming processes, and security gaps due to lack of access control.

For many small and midsize businesses (SMBs), it's not a matter of if they will need enterprise resource planning (ERP) software, it's a matter of when they will need it. As the company continues to grow, managing all that information across platforms becomes costly, time-consuming and prone to mismanagement.

Odds are good that a business relies on various software integrations to streamline data and access it cross-departmentally. But integrating multiple applications isn't cheap.

Between license fees, staffing, training, operations and the labor required to get them to work well together (if they even work together at all), the price of owning and integrating multiple pieces of software can add up real fast.

So Enterprise resource planning (ERP) systems are used by organizations looking to manage their business functions within a centralized and integrated system. ERP is commonly used by companies working within the supply chain to help keep track of all the moving parts of manufacturing and distribution. However, ERP can be utilized by a number of different industries including those in healthcare, nonprofit groups and hospitality. Organizations needing to manage their staff, customers and inventory can all rely on ERP benefits.

# Chapter 2

## ERP

We now get the problem of integrated complex systems and how it will affect our business if not dealt with in the right manner, they need to be managed in an efficient way and handle the flow of data. We also introduced an ERP solution to solve this problem but what exactly is an ERP? and how will it help?

**ERP** stands for “Enterprise Resource Planning” it’s a management software that integrates core business processes of a company by maintaining a common database.

A company usually consists of multiple departments and each department has its own functions processes that are performed within the region of this department but there exist the integrated business processes that companies use to perform their work which cut across the departments. ERP provides a way to manage these integrated processes from beginning to end by facilitating information sharing and connection between these departments.

ERP systems usually group common Function areas into so-called business Module like

**Financial management** This module manages your capital inflow and outflow. It covers standard Accounting & Finance transactions like ledger, balance sheet, tax management, and payments. The module also generates financial reports for different departments and business units.

**CRM** This module helps you to boost customer service and, eventually, profit. It manages leads, opportunities, and customer issues. Likewise, it provides a 360-degree profile of your customers by consolidating data like their social media activities, purchase history, and past interactions with support reps.

**Sales & Marketing** This module handles sales workflows like sales inquiries, quotations, sales orders, and sales invoices. The Sales and CRM modules work together to speed up the sales cycle and earn

the company more profits.

**Manufacturing** This module is sometimes referred to as Engineering or Production. It helps businesses make manufacturing more efficient in areas, such as product planning, materials sourcing, daily production monitoring, and product forecasting.

ERP job is to coordinate between the various systems that the business needs and make seamless integration with a shared database between them, hence eliminating common problems like

- No real-time data access for other departments.
- Latency to get information.
- More than one department having a need for the same data will cause high cost as data maintenance will increase.
- Shared data will need to be synchronized or a department will think it's lacking or having more than its needs.
- Moving information from one department to another is more prone to errors.
- Numerous disparate information systems are developed individually over time.
- Integrating date takes time and money.
- Inconsistency and Duplication of data.
- Lack of timely information leads to customer dissatisfaction and loss in revenue.

# **Chapter 3**

## **The ERP**

The ERP is the name of our implementation of an ERP solution and using this name will always refer to our implementation.

The ERP has two main components

- The main system which handles the system requests, controls the flow of data, and handles the users' data.
- The system Modules that handle the business processes, We have a variety of modules to handle business needs.

### **I Main System**

Here we will talk about The ERP Features, Development, and internal structure and explore each component in the structure. Let's begin with The features

#### **(1) System Features**

The ERP is Open source, cross-platform, and cloud-based, so users can use it from homes, so no need for the companies using it to have huge buildings or offices to hold down their employees. This will help startups to push their businesses out without the need for huge financial support. Also, The ERP is multi-organizational, meaning that multiple organizations can use it at the same time. Finally, The ERP is very modular in design and can be expanded easily due to its modularity and simplicity.

#### **(2) Development**

The ERP server and main functionality are written in C# based on the brand new open source and cross-platform framework Asp.net core developed by Microsoft.

Modules services are written in C++ and compiled as DLL files and then gets loaded by the ERP when needed at runtime. Client-side is mostly written using Angular framework with some libraries like Bootstrap and JQuery. For Data

storage we used Microsoft SQL server and MySQL database management systems and tools like swagger for API documentation. Next, you will find a table summarizing the tools used for development. Let's talk a little bit about the main tools we use and why we chose them.

### Asp.net core

Is a cross-platform, open-source, high-performance framework for building Cloud-based, Internet-connected applications. With ASP.NET Core, you can

- Build web apps and services, IoT apps, and mobile backends
- Use your favorite development tools on Windows, macOS, and Linux
- Deploy to the cloud or on-premises
- Run on .Net Core or .Net Framework

### Why

Before we list the reasons, we'd like to point out that the main goal or reason is learning new stuff and get the work done by this stuff

This framework is not mature enough and there is not that much of a community using it to help us when we face problems but we accepted the challenge and went on with it

Here are some of the main reasons

- It's a redesign of Asp.net 4.x with architectural changes that result in a leaner and more modular framework
- Open-source
- Cross-platform
- Easy integration of the client-side framework
- MVC easily implemented
- Built-in dependency injection
- A lightweight, high-performance, and modular HTTP request pipeline
- Ability to host on IIS, Nginx, Apache, Docker, or self-host in your own process (Kestrel)

### DLL

Stands for Dynamic Link Library and it's a Shared Library that can be used by more than one program at the same time and can be loaded into the program while running. By wrapping each of The Erp Modules services in a DLL file, we can

modularize our software and load them when needed only, this will help us save a lot of resources and enable us to have the ability to sell each module, maintain them and apply repairs, or upgrade to them separately and the system functions will not be affected.

### (3) Tools

Tool	usage	Requirement	Programming language
Asp.net core	server and main functionality	net core 2.2.x run-time Or SDK	C#
Angular	Client-side development	Node.js 8.12.0 Angular CLI 7.3.8	TypeScript Javascript CSS HTML
Entity framework core	An ORM to deal with the database	shipped with the framework	C#
DLL	Contain modules and main system services	C++ compiler	C++
Microsoft SQL server	Data Source to store system data	Microsoft SQL server	SQL
MySQL	Data Source to store system data	MySQL	SQL

Table 3.1: summary of the ERP tools used for development

### (4) System Structure

The main components of the system are shown in the figure 3.1 which will be explored in detail in the coming sections

#### System Main Components

##### API

Stands for Application Programming Interface, is an interface that simplifies the access to one's application services from another application, so it's the gateway to The ERP Services.

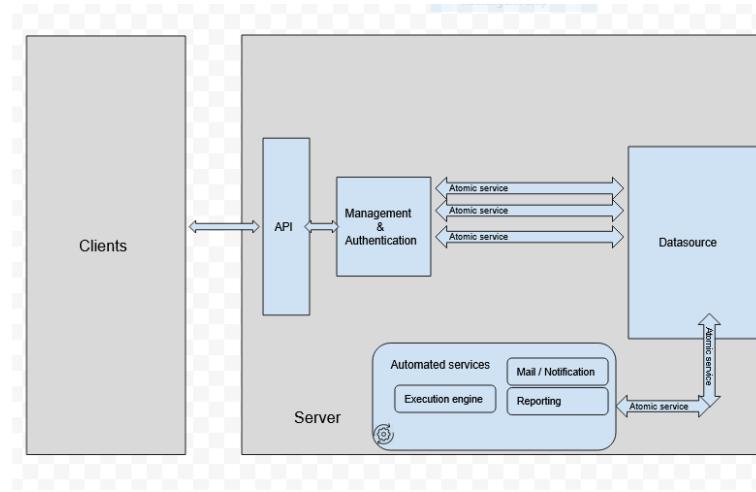
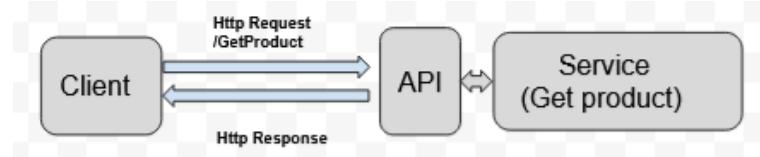


Figure 3.1: ERP System Structure

For web-based API as in The ERP, each service will have a URL like <http://www.TheErp.com/app/GetProducts>, and in order to use this service an HTTP request will be sent to this URL with some data (if required by the services), then the service gets invoked, and then return the result in the form of HTTP response.



The ERP API is well documented using the swagger tool, so any external system that is interested in using The Erp services can use this documentation.

### Role Management & Authentication

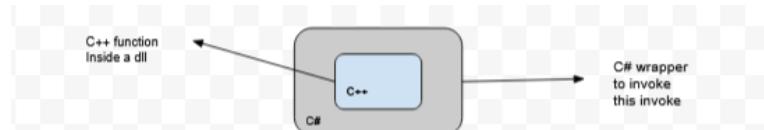
It is used to limit the access for certain services in The ERP. First, it Authenticates the incoming requests, in case of browser we use cookie Authentication; otherwise, Jwt Authentication is used, after authentication we can find out some info about the invoker like username, user id, roles (Administrator, man-

ager, ..etc), and company name because like we said before, The ERP can be used by multiple organizations at the same time, so the company must be included. The reason why we can find this info, is after a login request from a client has been validated, this info gets encrypted as a cookie in case of a client browser or a token otherwise, and sent back with the HTTP response to the login request, so that when a client calls a service which requires Authentication, the client must include the cookie or the token it received after successfully logged it in the HTTP request and then it gets decrypted and the information can be deduced.

## Services

First of all, let's define the term service. A service, in the case of The ERP, is a job that we want The ERP to do it for us, like getting an order, sending an email, or giving a report of the financial situation ..etc. If we look deeper into these jobs we can find out that some of them can be divided into smaller jobs. For example, getting an order, it first checks that the products in the order are in stock then accepts the payment, if they are, then send the order; otherwise, orders the manufacturing to start to produce this product and inform the customer. These smaller jobs are called, in our system, Atomic services because they can't be divided into smaller jobs.

These Atomic services can be used as a building block for other big services, such as in the case of sending an order. And it can be used as a service itself. So the bottom line is that The Erp is only containing services that are atomic in nature. Above that, the services are a composite of these Atomic services.



If we zoom in on atomic services we will find out that they consist of C++ functions and C# wrapper to serve as an interface to the C++ code.

Another kind of the services The ERP provides is the Auto-



Figure 3.2: Execution Pipe

mated Services . Automated services are the kind of services that run on their own, they do not need an invoker to run and The ERP is filled with them, below we list the important ones and their jobs

- **Real-Time Data**

This automated service is continuously monitoring the database for changes and if there is any, it will update the users with these changes to keep the data they have intact.

- **Notification System**

It's connected to all active users. Continuously sending notifications to them about their assigned Task, monitor the database for unsent emails then sent them.

- **Reporting**

Its job is to produce reports. For example, in case of a problem (connection to the database server or the manufacturing machines failed,...) it will try to contact any of the administrators by sending an email or an SMS to notify them about the problem to deal with it.

- **Execution Engine fig 3.2**

### **Execution Engine**

It consists of two pipes (Execution and Response). A task

is put into the execution pipe then pulled by the executioner to be executed. If a problem occurs (failed to connect to the database server) during execution, the executioner will then put the task back in the execution pipe for another trail.

If the execution produces a result, like for example (GetCustomer), it will be put into the response pipe to be sent back to the invoker of the Task and so on.

This helps when a problem led to make the services requests through API call unable to be invoked. For example, failed to connect to the data source, so the solution is instead of invoking them when requested through the API, which will lead to failure, we put them into the Execution Queue and we know the rest.

Another big use is when the service requested through the API is not atomic or it's atomic but Time-consuming, in that case, the services can be put one by one into the execution pipe and it will be taken care of. This will make the system more responsive.

## II The Modules

The modules are interconnected with each other and often have jobs to that each will have a share in like and they are integrated using the ERP.

The modules can be added separately to give users the customization and the choice of their business needs without overwhelming them with unneeded modules.

Our modules including, but not limited to, as we are working on new modules

- CRM
- Warehouse Management
- Accounting
- Manufacturing

## (1) CRM

CRM (Customer Relationship Management) is a strategy to manage the relationships of the company. It makes communicating with customers easier and improves marketing.



### Main features of CRM

- Customer Management
- Sales Management
- Reporting

### Customer Management

We collect and manage customer information and their interests.

- Qualified customers (Who have made orders)
- Non-Qualified customers (Who are considered to be the lead or target of the sales team)

Sales team communicates with non-qualified customers to make deals according to their interests of products.

If the customer accepts the deal, an opportunity, that has some products of customer's interests, is created.

### **Sales management**

We manage the sales through a pipeline of opportunities, each opportunity goes through 5 main stages which are (New, Qualified, Proposition, Negotiation, and Won).

When an opportunity is created, it is in New stage, then a salesperson communicates with the customer to make a deal with them. If the customer accepts the deal, the opportunity will be changed to Qualified.

After that, a salesperson proposes some products that the customer has an interest in and negotiates about their price. If the customer accepts to buy this product, the opportunity will be changed to Won and will be created as an order in the system's database.

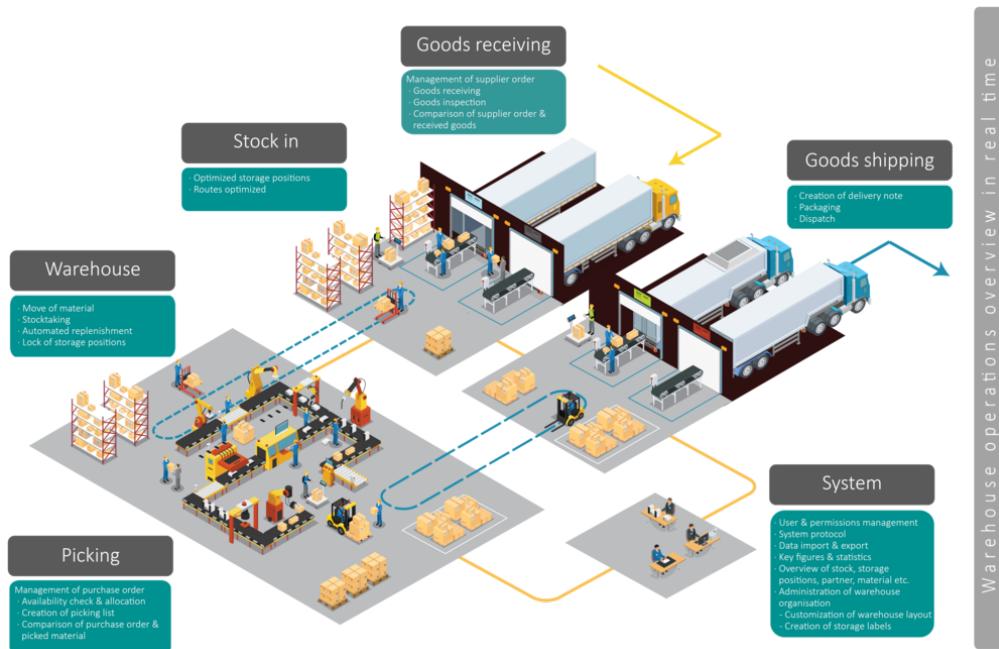
### **Reporting**

With CRM, salespeople and managers have access to dashboards where they can easily view important opportunity KPIs, in the form of graphs, charts, and more. They can also export data and create reports. This allows employees to analyze opportunities, and thus gain insights into sales and marketing activities. You can also analyze lost/won opportunities to improve your conversion rates.

## (2) Warehouse management

Warehouse management refers to the various processes related to maintaining and controlling a business' warehouse, such as the shipping, receiving, put-away and picking of goods. Warehouse management is responsible for everything that happens in a warehouse, whether they own one warehouse or several.

Managing inventory in warehouses with pen and paper can be inaccurate, slow and a lot of work, that's why it's better to use a Warehouse management system.



### Warehouse management system

Warehouse management system (WMS) is a piece of software that controls, records and automates various warehouse operations. The goal is to increase the overall productivity and efficiency of a business' warehousing operations.

#### What about our implementation of WMS ?

Our WMS can be used as a standalone system or can be integrated into The ERP. It supports multiple inventories, keeps track of products' moves, provides constant reports about orders delivery time, and has many other features.

## Features

Uses a database configured to support warehouse operations, containing details describing a variety of standard warehouse elements including

- Individual items that are handled and stored using weight, dimensions, automatic ID labels (bar codes, etc.), number of units in inventory, location in inventory, and whether it's sold or purchased
- Warehouse storage location and size
- Orders that are handled using automatic order number, customer's information, payment details, and time and date details (order being made, order ready for shipping, and order delivery to the customer)
- Ordered items with their quantities
- Supply requests with their time and date, suppliers' information, and the number of units to be supplied

Is responsible for

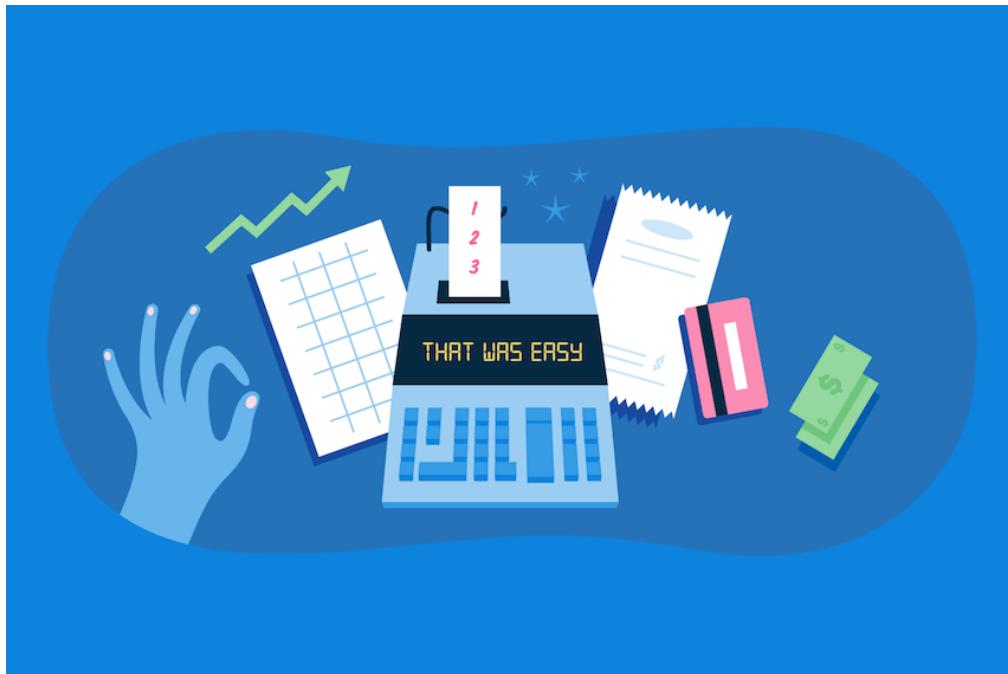
- providing Status of item quantity in terms of in stock, ordered, and ready for shipping quantities
- Handling Multiple warehouses
- Following up with each order throughout its stages from being made until it's ready for shipping
- Keeping track of products going in and/or out of inventories
- Organizing - sequencing the orders to be picked
- Reporting, which helps analyze the performance of warehouse operations, and find areas to improve
- Asking for supply when an item is out of stock
- Sending notifications when an order is completed

## Benefits

- Track locations, transaction histories, and costing
- Speed up the process of getting products into and out of inventory by keeping records of the location of each product inside the inventory

- Organizing - sequencing the orders to be picked helps Minimize the need for dock staging space, by having orders arrive at the shipping dock in trailer load sequence
- Proper reporting mechanism of inventory helps the manufacturing department in planning their future production schedules accordingly
- It helps track the payments and flow of finances into your Accounts

### (3) Accounting



#### Why Accounting ?

The financial module in ERP Systems provides financial functionality and analysis reports for balance sheets, ledgers, and financial statements.

Financial module is essential for an organization. Other modules cannot function without it. Successful implementation of financials is highly required for an organization in order to be trusted. All these factors explain the fact that financial modules are taken up first.

The finance module of an ERP system has the following sub-systems.

#### Accounting Subsystems

##### **Financial Accounting**

The objective of a good Financial accounting system is to provide control and integration of financial information that is essential to strategic decision making. The Financial accounting module of an ERP System gives you the ability to track financial accounting data within an international framework

of multiple companies, currencies, and charts of accounts.

### **General Ledger**

General Ledger is essential to both, the financial accounting system and strategic decision making

### **Accounts Receivables**

The Accounts Receivable Modules helps in tracking all the invoices that are awaiting payment from customers

### **Account Payable**

Accounts Payable Module (AP) — provides the functionality to enter, monitor, maintain, and process for payment of invoices and credit notes that the organization received from its vendors

### **Asset Accounting**

- Fixed asset management like acquisition, depreciation, retirement, etc
- Legal consolidation

A Legal consolidation process is a financial process that allows showing the assets, financial position, and income of a group as if the group were a single enterprise

### **Controlling**

Controlling enables the possibility to plan the financial parameters of the company and offers both, proactive capabilities for early warning if they become negative and complex analysis tools to identify factors of influence

## **Accounting Processes**

### **Budgeting**

Analysis of allocations, expenditures, revenues

### **Cash management**

Cash flow analysis  
What-if analysis

### **Capital budgeting**

Evaluation tools: NPV, IRR, pay-back period

## Our Module Features

- Show Company Accounting data
  - Show Sales (The products sold and how much the company earned)
  - Show Bills (Show the bills that the company has to pay and how much money is it)
  - Show Payments (Show the payments the company made, buying products materials or paid for bills)
  - Show Company invoices (For the payments have been made, if there was any)
- Show Customers Accounting Data
  - Billing data (Show the bills that the customer has to pay for the company)
  - Invoices data (Show the customer's invoice for each payment for the order he has made)
- Show Real-time data
  - To show data as soon as it changes with no need to refresh the page
- Reporting
  - Reports about profits, losses, and balance sheets

#### (4) Manufacturing



In order to sustain in the stiff competitive market, manufacturing companies invest a lot of time managing their diverse business functions rather than focusing on core business. This eventually leads to less productivity and profitability in the business sector.

A manufacturing management software comes with an ample number of features that helps in streamlining business operations, effectively transacting information across all business sectors, ultimately focusing on enhancing the business workflows.

Manufacturing is the process by which raw materials are transformed into finished products.

#### Manufacturing Module

The role of ERP manufacturing module is to complete the inventory management by implementing the operations specific to a streamlined manufacturing process. For a company, which handles a large number of manufacturing products, they need to track every manufacturing order efficiently and effectively.

#### Our implementation of Manufacturing Module

The Manufacturing Module in our system is independent and robust in handling the complexity of Production, managing bills of materials, planning the manufacturing Orders, etc. Manufacturing module is one of the basic applications in our system. Since the Manufacturing module is highly integrated with Inventory Management, you can keep your inventory automatically updated with each manufacturing process.

Our implementation of the manufacturing module helps businesses by dealing with the following subjects

## Products

Manufacturing module gives you the ability to add your manufacturing end products and determine a bill of materials for each product by using a friendly user interface.

### Bill of materials

Bill of Materials (BoM) is the basic building block of any manufacturing process. It is the list of raw material needed to produce a product. So while creating a manufacturing order for a particular product, one needs to select corresponding BoM. BoM will help the user to keep the inventory updated during the manufacturing process.

BoM allows a flexible environment for custom builds without starting from scratch.

### Manage Production

Once you have created and confirmed a Manufacturing order, you can start production. Our system will list all the Manufacturing orders.

After making your order, our manufacturing system will place your orders in a list describing the status of your order according to the inventory management system, starting date, and the current state of your order (done, canceled, confirmed).

### Robust Inventory Tracking

Manufacturing plants deal with raw material and finished goods inventory. Raw material management is the process of keeping track of all appropriate material required to ensure that the business carries on uninterrupted manufacturing processes. On the other hand, finished goods inventory includes products that the manufacturing plant has produced, and they need to be managed to keep track of how and when they would be transported to the warehouses or customers.

This is where many manufacturing modules experience challenges, as these two processes must be synchronized to avoid inappropriate or insufficient production, which would bring customer dissatisfaction, and cause losses to the business.

Keeping watch on these manually is quite impossible. For this, you need an ERP software that can track complete inventory. The automated application would help reduce human errors and improve inventory management like raw materials re-ordered needs or track the delivery dates of the finished goods, etc., thus keeping your manufacturing running seamlessly.

From that context, our manufacturing system has a robust inventory tracking

since you have placed your order and in case of lacking raw materials, manufacturing module will change the status of your order to waiting, until the raw materials are available. Our manufacturing module will track these changes and update your order status.

### **IOT Box**

IOT Box is one of the important features that our system offers, As it provides a way for the system to communicate with hardware components directly. This will give the business even more managing capabilities by integrating the hardware in the system to monitor the hardware and the process it does.

IOT box is a physical box that we offer that can be connected to our system directly and the box can be then connected to the business hardware, like physical machines, to do its job.

The current IOT box is limited in features as it will send commands to the box that can in return send them to the hardware we want to control. It works as a microcontroller that can be programmed from the system to do its functions. The IOT box is tended to have more capabilities and have different ports added to it to be used for different activities.

**(5) Product Manager**



Product manager do manage products

# **Chapter 4**

## **System Access**

### **I The ERP**

We wanted to achieve, with The ERP, the ability for the users to be able to use the system from anywhere to give the users more diversity of the places they can check the system from or achieve tasks and also give the users the ability to determine what can be achieved and from where to satisfy business rules and security.

To achieve this. The ERP comes with more than one plan, the first is to have the code for the program and use it within the company.

To achieve the more diversity options, The ERP will be on the cloud for different businesses to use and benefit from it, which allows users to access it from anywhere and also with a mobile app to have a more concise view of the system.

### **II Mobile Application**

#### **(1) ERP App Main Task**

The main task of the ERP app is to provide an easy way to monitor the modules of the ERP system

#### **(2) Main Features**

- Authenticate admins
- Authenticate users
- Rest API with the main ERP server
- Display all modules
- Display statistics of each module

### (3) Application Screens

- **Authentication sign up screen**

Responsible for registering the full data needed of the customer or the admin of the module.

- **Authentication login screen**

Responsible for getting email and password from the user for authentication. Get the user type if admin or customer.

- **Modules Screen**

Displays all modules

- **CRM Screen**

Displays more information about the module and some statistics about the customers

- **Billing & Accounting Screen**

Displays more information about the module and some statistics

- **Warehouse Management Screen**

Displays more information about the module and some statistics

- **Profile Screen**

Displays the current modules and some settings of the application

The ERP application is made with flutter which is Google's mobile app SDK, complete with a framework, widgets, and tools, that gives developers an easy way to build and deploy visually attractive, fast mobile apps on both Android and iOS platforms.

## **Part II**

## **BPM**

# Chapter 5

## Introduction

With this new part, we will get to discover a new way to approach business processes and help improve our business, it has its own unique way that is different from ERP approach but to get BPM approach we will need to understand what a business process is.

A business process is an activity or set of activities that can accomplish a specific organizational goal. Business processes should have purposeful goals, be as specific as possible, and have consistent outcomes. It is a collection of activities that takes one or more kinds of input and creates an output, such as a report or forecast, that is of value to the customer.

ERP software supports executing business processes by supporting the efficient operation of business processes by integrating tasks related to sales, marketing, manufacturing, logistics, accounting, and staffing—throughout a business. In addition to this cross-functional integration, which is at the heart of an ERP system, companies connect their ERP systems, using various methods, to coordinate business processes with their customers and suppliers.

So how does BPM come in the picture? Business Process Management (BPM) is an approach that focuses on capturing and improving business processes to make an organization more efficient. This can be achieved by first capturing an organizations' current-state end-to-end processes and then documenting the steps in process maps.

While the ERP job cares more about cross-functional integration and tends to be limited to organizational functions, BPM job tends to be much more process-focused.

Most companies, when undertaking a business improvement initiative, will consider implementing a dedicated BPM system to help them model, analyze, and optimize processes to drive the business transformation forward.

BPM tends to be flexible with approaching business processes design. It can manipulate, change, fix and monitor a business process easily. The usual cycle

of BPM fig 5.1 helps to achieve this flexibility which includes

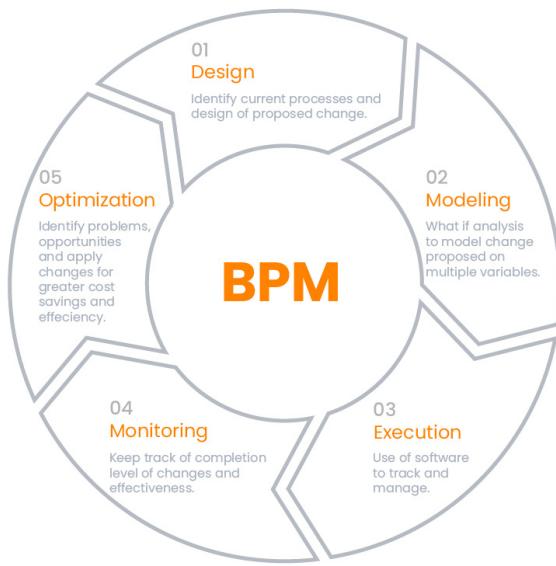


Figure 5.1: BPM Cycle

- **Design** Process design encompasses both the identification of existing processes and the design of future processes. Areas of focus include representation of the process flow, the factors within it, alerts and notifications, escalations, standard operating procedures, service level agreements, and task hand-over mechanisms
- **Modeling** Modeling takes theoretical design and introduces combinations of variables. For example, changes in rent or materials costs, which determine how the process might operate under different circumstances.
- **Monitoring** Monitoring encompasses the tracking of individual processes, so that information on their side can be easily seen, and the status of their performance of one or more processes can be provided. In addition, this information can be used to work with customers and suppliers to improve their connected processes and measures information of three categories: cycle time, defect rate, and productivity
- **Optimization** Process optimization includes retrieving process performance information from modeling or monitoring phase; identifying the potential or actual bottlenecks and the potential opportunities for cost savings or other improvements; and then, applying those enhancements in

the design of the process. Process mining tools are able to discover critical activities and bottlenecks, creating greater business value

We mentioned that BPM helps capture end-to-end processes, but **what does end-to-end process really mean**

What end-to-end process really means is from start to finish. The goal is to understand and thus to assess and improve an entire process —not just its components.

## I What is a process ?

A process is a sequence of tasks. Every task is an action, and is carried out by a person or by some automatic system. All processes share certain characteristics

### *Interaction over time*

A chronological relationship between tasks, with due dates and sequencing but not a schedule.

### *Multiple actors*

More than one person or automatic system must complete tasks for the whole process to be successful.

### *Repetition*

The sequence of tasks is repeated, either at fixed intervals or when triggered by a specific event.

## II What Is Not a Process ?

The action of filling out a form is not a process. A form that has many pages and is filled out online is a single task for a single user. A form that can be filled out in more than one sitting is a single task for a single user. Even a form that has smart fields that depend on other fields, with conditional display, is a single task for a single user. A state diagram is not a process. A process is constructed from actions. The things that are updated by these actions might have states associated with them, so you might create state diagrams as part of your process validation, but the state diagram is not itself a process.

# Chapter 6

## The BPM

Yet again we have a name for our implementation of BPM that we will use over the next few chapters and it follows the same pattern for our naming schemes, “**The BPM**”.

The BPM is meant to work best when integrated with other applications to be used to its potential. It will provide all the tools to be used inside other programs. But it still works as a full Standalone BPM solution to design, execute and monitor Business processes.

It offers a pipeline that works on three stages each can work as a standalone application, the pipeline works as follows

- Design business process using a graphical application
- Execute the business process with the ability to modify its execution at runtime
- Monitor the business process and see the process execution in detail and change the execution graphically

### I The Process

For each process The starting conditions and timetable of a process. The following question need to be considered

Under what conditions will the process start ? These conditions were referenced from the book Process-Driven Applications [3]

- **Time**

- The process starts at a particular time
- The process starts after a particular time interval
- The process starts in relation to another time

- **Conditions**

The process starts once one or more conditions are met, for example, a warehouse is restocked with a particular product once stock levels go below a predefined threshold value.

- **Messages**

The process waits for a particular message to arrive.

- **Events**

Events have an important role, especially in exception situations. An event is triggered when an extraordinary situation arises that requires special handling, for example, a late delivery, a machine failure, an accident, or Similar.

After the designer answers these questions, he will change the setting of the starting node to satisfy the conditions required for this process.

For each process, there will be three start nodes, two of which will have their starting conditions embedded in them

- The first will be triggered when the process loaded in the engine can be used to initialize data related to the process or when integrated with a GUI application that can be used to generate a GUI for the process when loaded.
- The second will be when the process awakes and starts to run, whether it is its first time to run or awaken after a pause, can be used to get data to memory back before continuing the process.
- The third will be the one which will start based on the question that the user answered and can choose which event will make it start.

### (1) Involved Process Roles

The involved process roles define who is responsible for performing which activities during the process. All you need is a simple list of process roles and a description of the functions the roles perform within the process. At runtime, users and groups from the company's user management solution are assigned to these roles. Although this presents a challenge, the complex task of assigning users to process roles in workflow management systems within company organizations is outside the scope of this book and is not discussed in more detail here.

## II Notation

While designing our notation we came across two other notations that are used as standards like

**BPMN 2.0** [1] (Business Process Model and Notation), it provides businesses with the capability of understanding their internal business procedures in a graphical notation and will give organizations the ability to communicate these procedures in a standard manner.

**BPEL** [2] (Business Process Execution Language) is an XML based language that allows Web services in a service-oriented architecture (SOA) to interconnect and share data.

Our notation is affected deeply by BPMN 2.0, but our goal was simplicity so we removed some of its complexity and added some other nodes that can be used in an easy and intuitive manner, So while BPMN 2.0 is much capable of executing complex business processes. Our notation is much more simpler to use and understand.

### (1) The Nodes

They should be descriptive enough to understand business process like interaction with users and repetition of tasks, simple to be intuitive to use and have the ability to be extended for advanced usage.

#### Start



Specifies the start of the flow.

Each flow will have three start nodes, OnLoaded - OnAwake - OnStart.

The latter must be explicitly defined.

The others will be implicitly defined to do nothing if not explicitly defined.

**End**

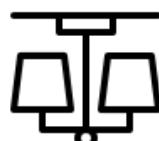
Specifies the end of a flow or just a branch from the flow.  
Each flow can have one or more than one end node.  
Each will be the end of one flow or more than one.  
You can specify logging for each end to specify what ended. defined.

**Service Task**

Executes Web services based on REST.  
Will Create the request and handle the response. defined.

**Database**

Works with an SQL database to connect, modify and get data.  
Can work with different SQL implementation (Oracle - MySQL - SQLite).  
Can listen to the database for data insertion using polling.  
Provides the user with a way to visualize the database while choosing a query to execute, like showing the table and data. defined.

**Parallel**

Makes more than one flow to be executed in the same process.  
The two flows will be executed at the same time in the same environment using parallel processing. defined.

**Condition**

Divides the flow into more than one flow, each with a condition.  
When the flow reaches this node, it will evaluate the condition to which flow an will choose the flow that evaluates the condition to true. defined.

**External Event**

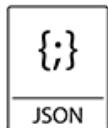
Executes Events from outside of The BPM nodes. defined.

**Timer Event**

Waits for a certain time before proceeding with the flow.

**Script**

Executes a predefined script written by user to add functionality to The BPM.

**Json**

Creates a JSON string or get the values of the elements in a java string.

**Increment Function**

A function that increase variable by 1 used in loops

### III Tasks

Tasks are the main components of The BPM. Each business process will have a variety of tasks that need to be executed. The BPM supports different kinds of tasks.

- **API Tasks**

They are tasks that are defined through an API which is a defined interface of how to interact with other applications functions. It can be used to add functionality to The BPM by executing functions available at other places

- **Script Tasks**

They are tasks that are defined through an API which is a defined interface of how to interact with other applications functions. It can be used to add functionality to The BPM by executing functions available at other places

- **User Tasks**

A User Task is used to model work that needs to be done by a human actor like filling a form or accepting a request.



Figure 6.1: User Task

- **External Tasks**

They are tasks that are only known to the business, and they have no defined way of accessing them from outside of the system, so the only way to execute them is by executing them inside of the system, So The BPM will define a way to define the tasks by the names known to the business and will put them in a Queue waiting for the system to look for them and get them then execute them. Fig 6.2

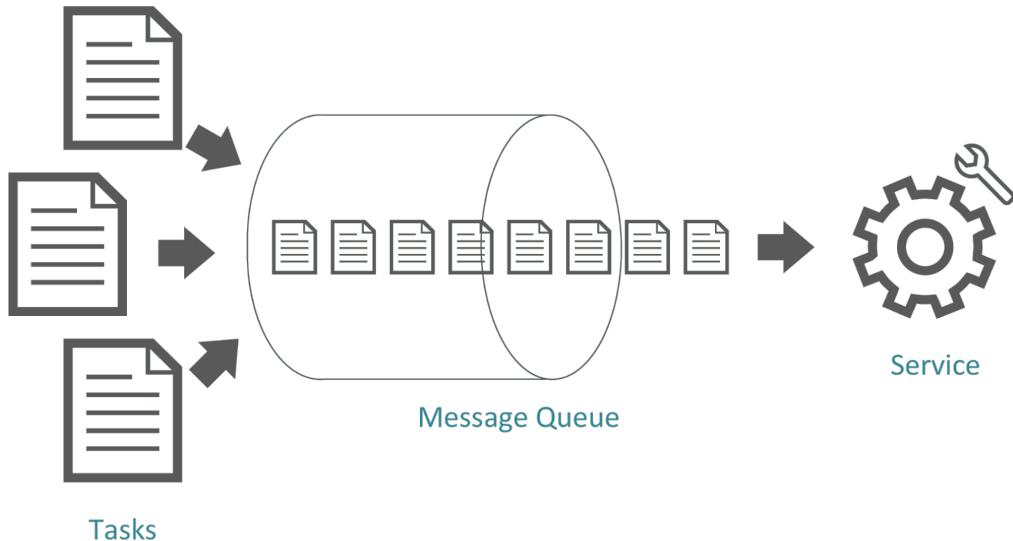


Figure 6.2: Tasks Worker

## IV Variables

Variables are the way to store data in The BPM and are used to exchange data between different tasks and nodes.

Variables can be one of these data types

- Booleans
- Strings
- Numbers

Many nodes have input and output fields that can be assigned to a variable that can change how this node will behave.

## V The BPM Cycle

As stated before BPM works in a cycle and we will see how this cycle is executed in The BPM to execute business processes.

The first thing the business will need to decide the business rules after deciding on the business rules the business will use our suit of applications.

Each Application works as stand alone application to execute part of the BPM cycle

### (1) Designer

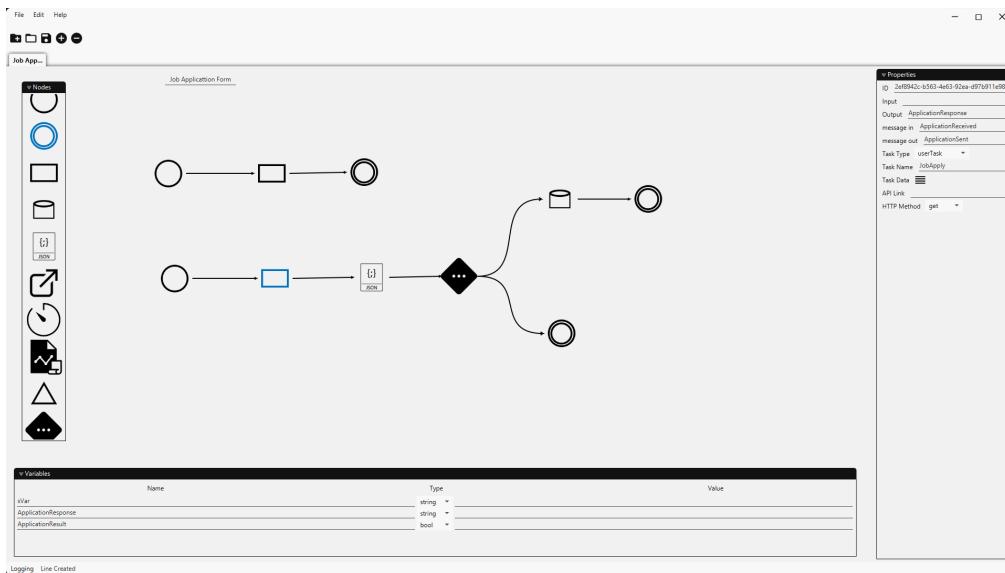


Figure 6.3: Designer

The designer is used to design the business process and adding variables to be initialized at run time and modify how each node will behave. It includes

- A SQL editor to format SQL query for Database node, request editor for service task node
- A JSON Editor to define request or objects in JSON format and can also be used to read
- A User task editor to define tasks that need to be done by human actors

The designer enables the user to design the workflow, add variables to be added to run time and properties window to change the nodes setting.

The process is then generated in an XML file format and the file is now ready for the engine to load and execute.

Tools used to build designer

- Java
- JavaFX 11 used to build the GUI as it allows you to create Java applications with a modern, hardware-accelerated user interface that is highly portable
- Gradle as a build system

## (2) Engine

The main system in the engine is the workflow engine which is used to execute the process by moving the flow of execution from one node to another.

A workflow engine manages and monitors the state of activities in a workflow, such as the processing and approval of a loan application form, and determines which new activity to transition to according to defined processes (workflows). The actions may be anything from saving an application form in a document management system to send a reminder email to users or escalating overdue items to management. A workflow engine facilitates the flow of information, tasks, and events. Workflow engines may also be referred to as Workflow Orchestration Engines.

Workflow engines mainly have these functions

- Verification of the current process status: Check whether it is valid to execute a task, given current status
- Determine the authority of users - Check if the current user is permitted to execute the task
- Executing condition script: After passing the previous two steps, the workflow engine executes the task, and if the execution is completed successfully, it returns the success, if not, it reports the error to trigger and roll back the change
- A workflow engine is a core technique for task allocation software, such as business process management, in which the workflow engine allocates tasks to different executors while communicating data among participants. A workflow engine can execute any arbitrary sequence of steps, for example, a healthcare data analysis

The Engine has an API that enables other applications to use its functions and integrate them into their systems and it is what the ERP uses to integrate The BPM inside it.

The API gives other applications the ability to manipulate workflow

- Load, Start, alter and stop execution
- Get the tasks

Tools used to build engine

- Java
- Gradle as a build system

### **(3) Monitor**

The Monitor is used to follow the execution of a process and can modify its execution, it helps give users the ability to view the process and monitor it if anything went wrong or if there is a bottlenecking anywhere in the process that needs to be modified and get an idea of how the process is going

Tools used to build monitor

.Net core to be integrated easily with The ERP application

# Chapter 7

## BPM Examples

### I Examples of Process-Based Applications

The following examples are familiar to many organizations.

We will start with a simple example that shows how to model a simple business process.

#### (1) Bakery

We want to model the following situation using BPMN 2.0. For a bakery which will start at the beginning of each day. They will start by baking the bread then selling it and at the end of the day, all the workers will receive their payments.

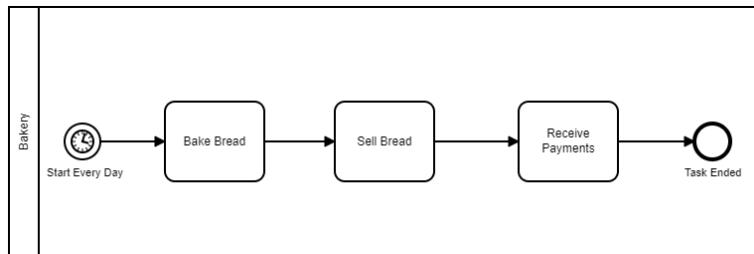


Figure 7.1: Bakery Worflow

For our Implementation

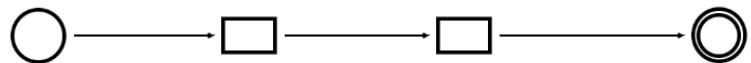


Figure 7.2: Bakery Worflow

## (2) Payment Request

We will use the following example to illustrate how to model a two-step escalation using BPMN 2.0. When we want a pizza, we order one. Sometimes the pizza delivery screws up and the delivery takes longer than 20 minutes. Then we complain to the delivery service. After that, we give them another 30 minutes to deliver the pizza. If they do not make it in time, we give up and cancel our order.

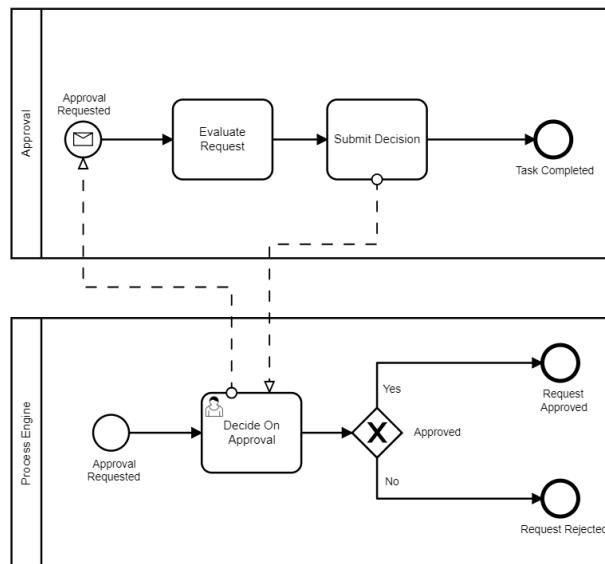


Figure 7.3: Payment Request Worflow

For our Implementation

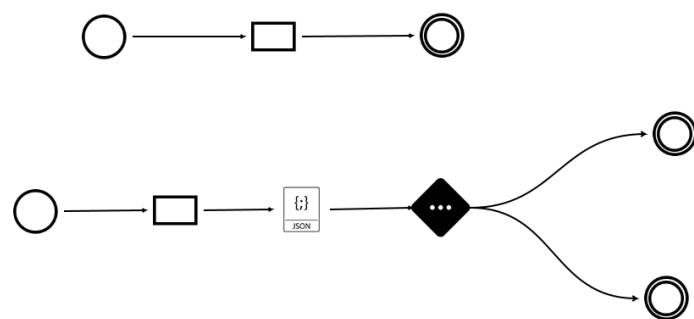


Figure 7.4: Bakery Worflow

# **Part III**

# **Hardware**

# **Chapter 8**

## **The Factory**

The Factory is a separate project that represents a business model with certain needs for the business to succeed and make profits.

The business model involves business processes that are the core of this business. These processes include

- Manufacturing the products
- Warehousing
- Provide an online market to show the products and sell them
- Delivering the products to the customers

The Factory is made as a prototype of a business that wants to include technologies in their business like

- IOT
- Factory 4.0
- Provide an online market to show the products and sell them
- Delivering the products to the customers

The main purpose of The Factory project is to show a migration of an existing business with an already working business model with The ERP and The BPM and how the business benefits by migrating the business with our solutions.

We will first discuss The Factory as a separate project, its components, and how it works. Then we will have a look at how migration can be done.

A flow diagram that shows how The Factory works in a separate matter Fig 8.1

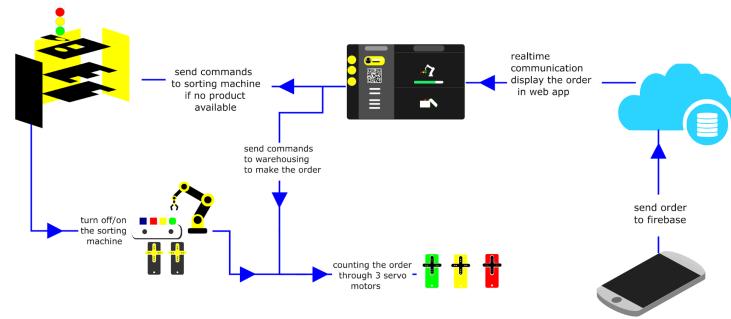


Figure 8.1: The Factory Flow

# Chapter 9

## Hardware

### I Factory box Design

We made all the factory box from black and yellow acrylic sheet with thickness 3 mm with a press fit design to make assembling easier.

First, there are many programs used to make 3d designs like SolidWorks or Inventor but we used CorelDRAW 2d program. It is a vector graphics editor that exports DXF files that the CNC machine uses to execute the design.

It consists of three main parts. Fig 9.1

- Sorting part
- Warehousing part
- Collecting part

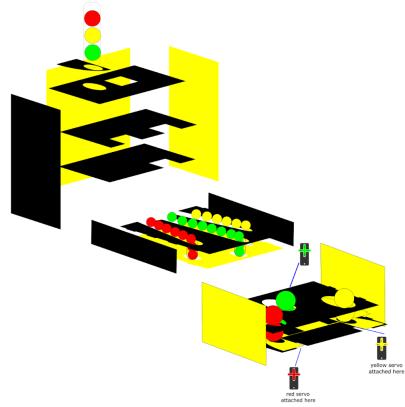


Figure 9.1: Factory Box

### (1) The sorting part

This part is considered the first stage of our factory. The aim of this part is to make sorting to our spherical products (colored ball) based on their colors then put them in a fixed place to bring them quickly.

The base of this frame is a (10 cm \* 15 cm) and the height is 30 cm.

It has 2 servo motors (upper and lower) and one color sensor. The upper servo takes the ball from the cylindrical tube to put it below the sensor, so the sensor detects its color and the lower servo moves the ball to the second stage in a specific route. Fig 9.2

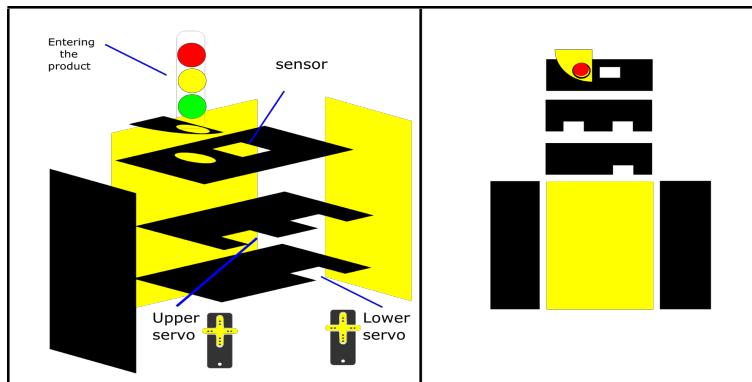


Figure 9.2: Sorting Part

### (2) Warehousing part

This part is considered the second stage. The aim of this part is to store the balls in a specific route based on their colors.

We made the route slightly leaning forward to make the motion of the ball smooth. There is a hole at the end of the route to drop it to the third stage.

The base of this part is a (20 cm \* 15 cm) and the height is 5 cm. Fig 9.3

### (3) Collecting part

This part is considered the third stage. The aim of this part is to take the product from the warehousing stage and drop it down to the delivery box. It has 3 servo motor as we have 3 colors in the warehousing, each servo motor moves with a specific angle to drop down the ball into the box and return to the first angle to take the second ball ..etc

The base of this part is a (20 cm \* 10 cm) and the height is 10 cm. Fig 9.4

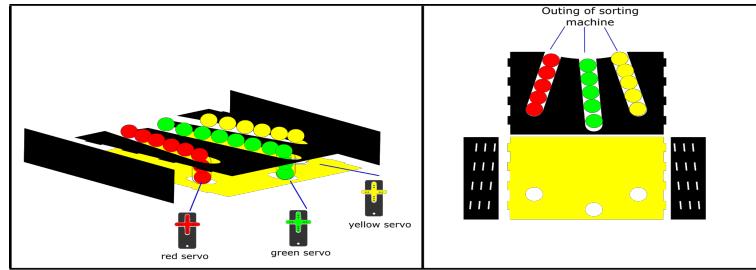


Figure 9.3: Warehousing Part

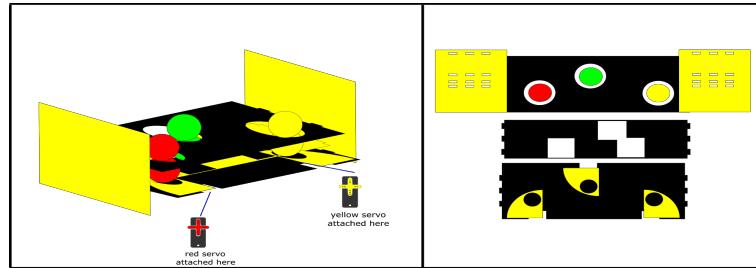


Figure 9.4: Collecting Part

#### (4) The Base

Finally, This part is made to collect the other parts on it.  
The base of this part is a (40 cm \* 35 cm) and the height is 10 cm. Fig 9.5

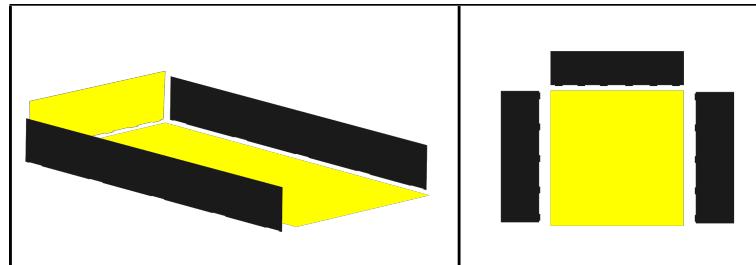


Figure 9.5: The Base

## II components used

- TCS230 RGB Color Sensor
- Servo Motor

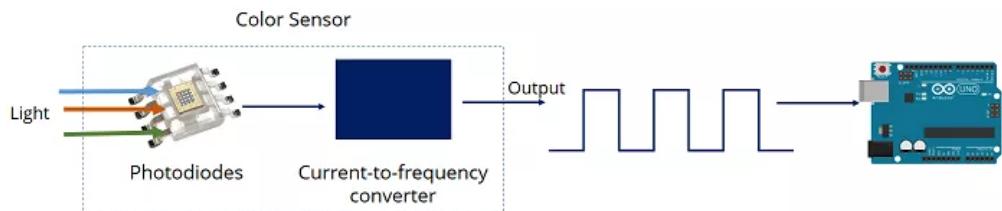
- Arduino Uno
- Node MCU

### (1) TCS230 RGB Color Sensor

The TCS230 senses color light with the help of an 8\*8 array of photodiodes.

Then using a Current-to-Frequency Converter the readings from the photodiodes are converted into a square wave with a frequency directly proportional to the light intensity.

Finally, using the Arduino Board we can read the square wave output and get the results for the color.



If we take a closer look at the sensor we can see how it detects various colors. The photodiodes have three different color filters.

Sixteen of them have red filters, another 16 have green filters, another 16 have blue filters and the other 16 photodiodes are clear with no filters.



The 16 photodiodes are connected in parallel. Using the two control pins S2 and S3 we can select which of them will be read. For example, if we want to detect red color, we can just use the 16 red filtered photodiodes by setting the two pins to LOW logic level according to the table.

The sensor has two more control pins, S0 and S1, which are used for scaling the output frequency. The frequency can be scaled to three different

preset values of 100%, 20%, or 2%. This frequency-scaling function allows the output of the sensor to be optimized for various frequency counters or microcontrollers.

<i>S0</i>	<i>S1</i>	<i>Output Frequency Scaling</i>	<i>S2</i>	<i>S3</i>	<i>Photodiode Type</i>
L	L	Power down	L	L	Red
L	H	2%	L	H	Blue
H	L	20%	H	L	Clear (no filter)
H	H	100%	H	H	Green

## (2) Servo motor

A servo motor is a closed loop servomechanism that uses position feedback to control its motion and final position. The input to its control is a signal (analog or digital) representing the position commanded for the output shaft.



Controlling a servo motor directly from the Arduino is quite easy. However, a servo motor may require significantly more current than the Arduino can provide.

So we used an external power supply to provide the required current to the servo motor.

We used 2 servo motors in the sorting part (upper and lower servo) and 3 servo motors in the collecting part (red, green, and yellow servo).

## (3) Arduino Uno

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header, and a reset button.



It contains everything needed to support the microcontroller; simply connect it to a computer with a

USB cable or power it with an AC-to-DC adapter or battery to get started.

We used Arduino Uno to run the sensor and to control the 2 servo motors of the sorting part

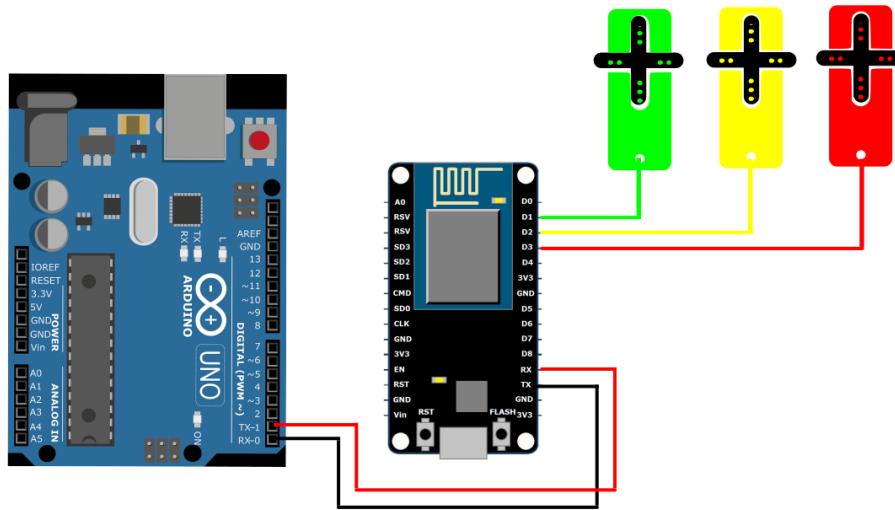


Figure 9.6: Sorting Connection diagram

#### (4) Node MCU

In our project, we need wireless communication to connect the Arduino to the web application, so we used NodeMcu.

It is an open source IoT platform. It includes firmware which runs on the ESP 8266 Wi-Fi SOC.



We also used it to control the 3 servo motors of the collecting part as it has PWM output pins.

We used serial communication to connect the Arduino to the NodeMcu to exchange the data between them.

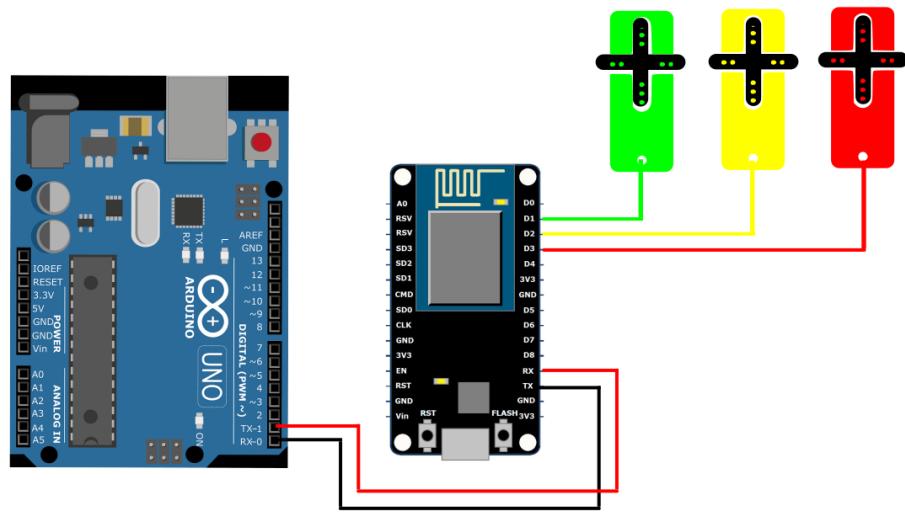


Figure 9.7: Collecting Connection diagram

# Chapter 10

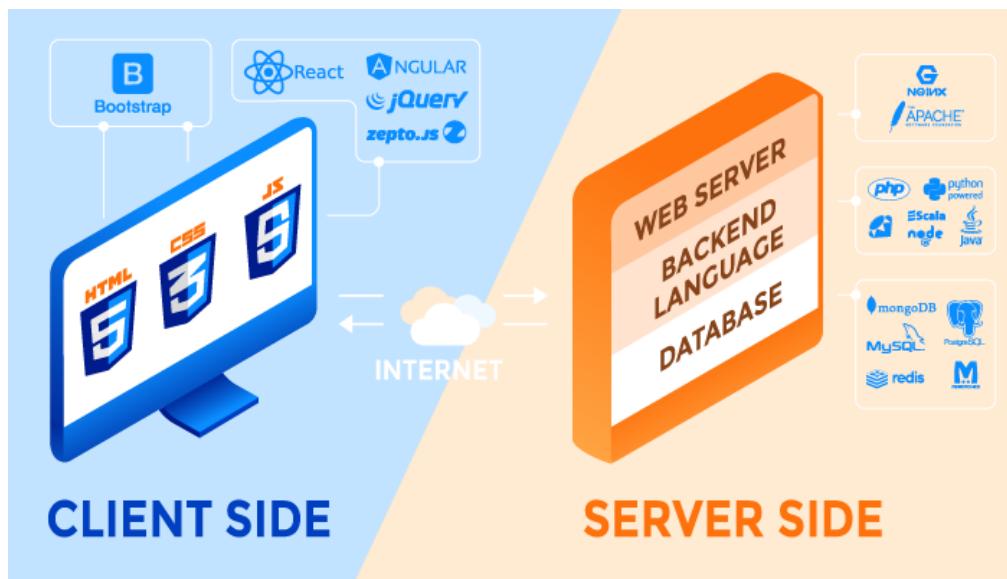
## Software

The software will provide a way to have control over the project and visual feedback of the system overall and offers the products for sale through an online market.

The software system is divided into three parts

- Server Side
- Client Side
- Mobile Applications

### I Server Side



A server is a computer program or a device that provides functionality for other programs or devices, called "clients". This architecture is called the client-server model. A server should be reachable from a user's local

computer, smartphone, or other devices. Operations may be performed in server-side because they require access to information or functionality that is not available on the client, or because performing such operations on the client side would be slow, unreliable, or insecure.

The Factory software is built upon this architecture, where the server provides an API to be accessed by the clients using it.

The server also talks to the hardware using Socket for real-time, bi-directional communication between clients (Node MCU in our case) and the server.

The server API provides

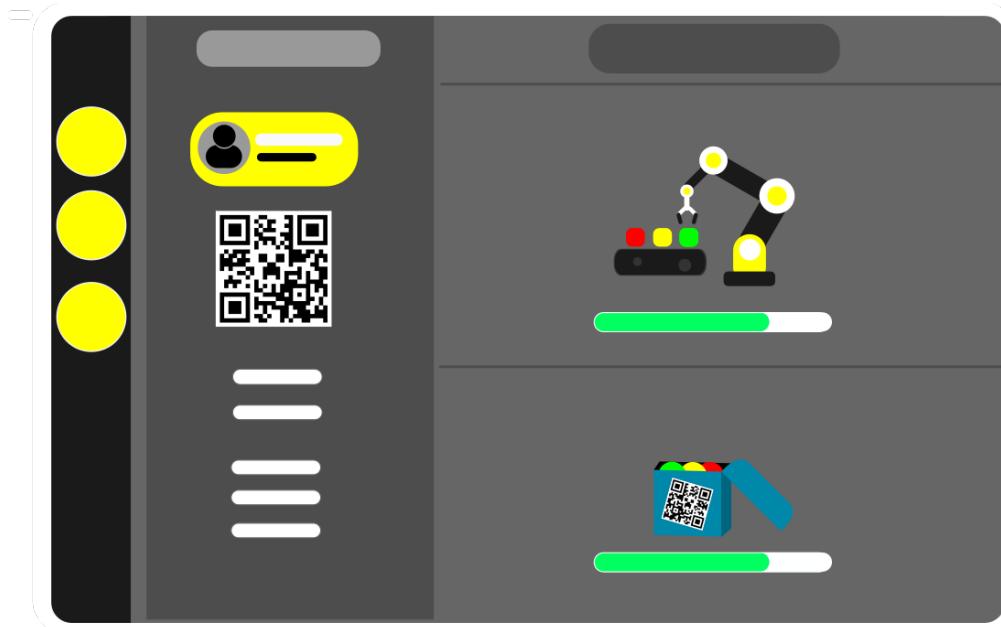
- User Authentication, it will validate the user signing in the system
- User Data, it will get the data related to the user
- Getting the orders
- Sending the order to the hardware

All connections to the database are done using the server. The database used is firebase, which is a cloud database solution that helps to quickly develop high-quality apps and grow the business. Data is stored as JSON and synchronized in real-time to every connected client.

The server uses the database to store data about

- Users of the system
- Clients
- Orders
- Drivers (Which we will discuss their purpose later in the book)

## II Client Side



The client-side refers to a computer application, such as a web browser, that runs on a user's local computer, smartphone, or other devices, and connects to a server as necessary.

Client side will provide a way for users of the business to log into the system, control the hardware and have an overview of the overall business, like the orders made by customers and the state of the orders.

The client app will offer these operations

- Display the orders in details
- Talk to the server API
- Sending orders to the hardware via server API. Sending order to hardware to start the manufacturing and prepare process, as orders aren't permitted to start without a user manually evaluating the order and then send the order to the hardware
- Monitor the order process. Monitor the order allows the process to have feedback on how the process is doing or if there is any problem with the system

The client app is built using Vue.js which is a progressive framework for building user interfaces and single web applications.

Vue.js has these features that will make the developing simpler

- Utilizes a virtual DOM
- Provides reactive and composable view components
- Maintains focus in the core library, with concerns such as routing and global state management handled by companion libraries

### III Client App

Smartphones are no longer just devices for calls, they are also used to send multimedia messages such as pictures, videos, and emails. Due to the enormous potential of smartphones, these capabilities can be exploited by multiple applications that benefit the user, so we tried to make simple and useful applications to our hardware project.

The aim of this client android application is providing an easy way to make an order from the factory and track the order until it reaches the customer's location. It also displays all the orders from the client.

#### Tools used

- Software Requirements: Android studio, Java
- Database: Firebase

#### Features

- Provides users authentication
- Makes orders
- Tracks order location by using google map
- Receives notifications about the order
- Checks the Qr code of the order
- Displays all orders
- Provides statistics profile

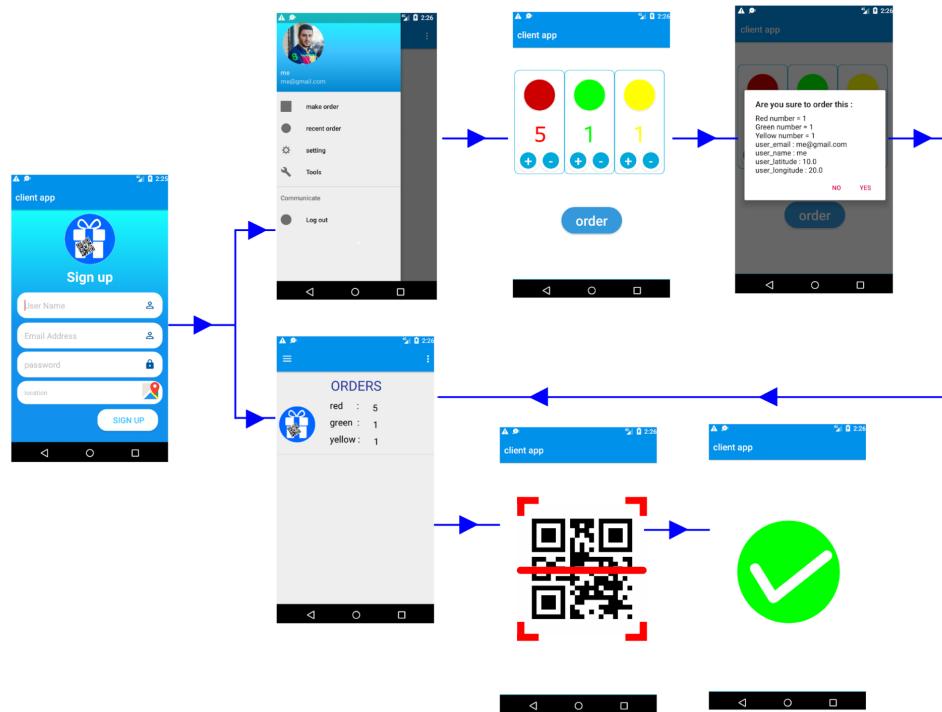


Figure 10.1: Process Flow of the Client Application

### (1) QR Code

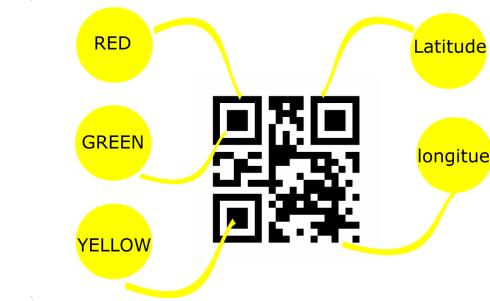
A QR code (quick response code) is a type of 2D barcode that is used to provide easy access to information through smartphones.

In our android applications, we need to store the order of each customer and the location of the customer.

We stored the data we need in a JSON object to extract the data from it in an easy way.

**For example**

```
{
    Red : 5,
    Green : 1,
    Yellow : 1,
    Latitude : 30.35,
    Longitude: 30.59
}
```



As the order consists of three products (red, green, and yellow) and the location of the customer (latitude and longitude) that our application uses to get the location on google maps.

**(2) Process Flow**

The process flow of the client application during making the order Fig 10.2

- The client makes an order
- The firebase sends the notification order to the web app to collect it
- The web app collects the order and puts it in a box with a printed QR code
- The delivery service takes the order and delivers it to the client
- The client scans the QR code of the order
- The client app displays the contents of the box

**IV Important screens**

- Getting the location of the client by using google maps In the registration form, when the client presses on the icon of google maps, the app will open google maps screen and store the location of the client
- Scanning the Qr code of the order when receiving it, then the application will display the status of the order if it is the right or wrong order

**V Driver App**

The aim of this driver android application is providing an easy way to get to the final destination of the order on google maps then deliver it to the client

**Tools used**

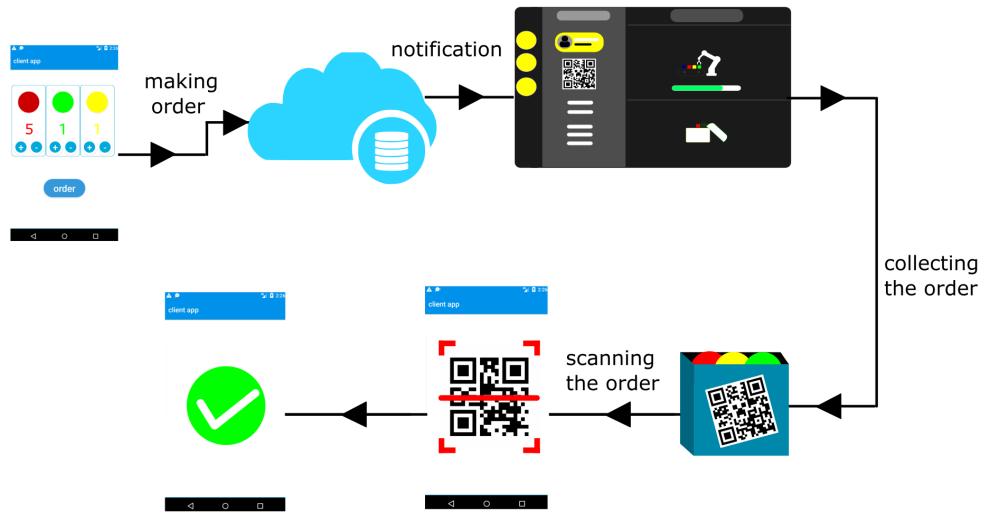


Figure 10.2: Making Order Flow

- Software Requirements: Android studio, Java
- Database: Firebase

### Features

- Real-time communication
- Users authentication
- Checks the QR code of the order
- Displays the location of the customer
- Displays all targeted locations

### (1) Process Flow

The process flow of the driver application during making the order Fig 10.2

- The web app sends a notification to the driver app after finishing the order
- The driver scans the QR code of the order
- The driver app displays the destination of the order on google maps and draws the shortest route between the two locations

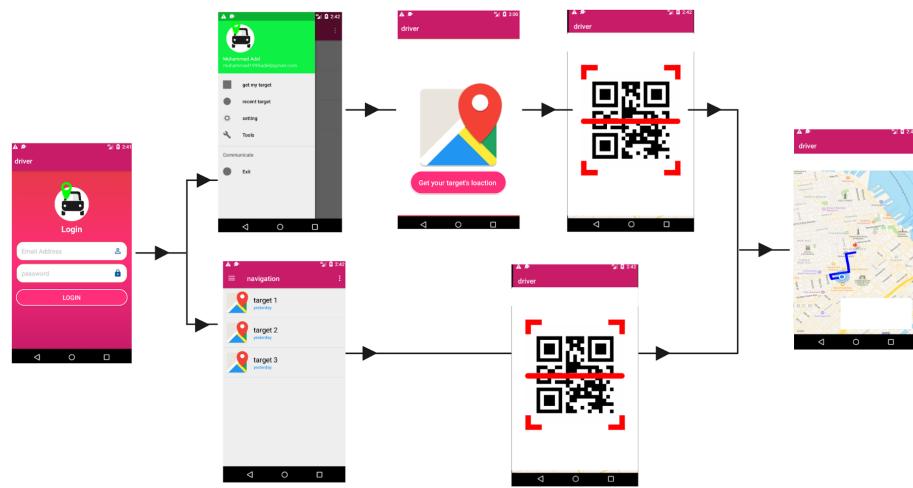


Figure 10.3: Process Flow of the Driver Application

## (2) Important screens

- Scanning the QR code of the order, then the application will get the latitude and longitude of the customer and pass it to google maps activity to display the shortest path between the driver and the customer.

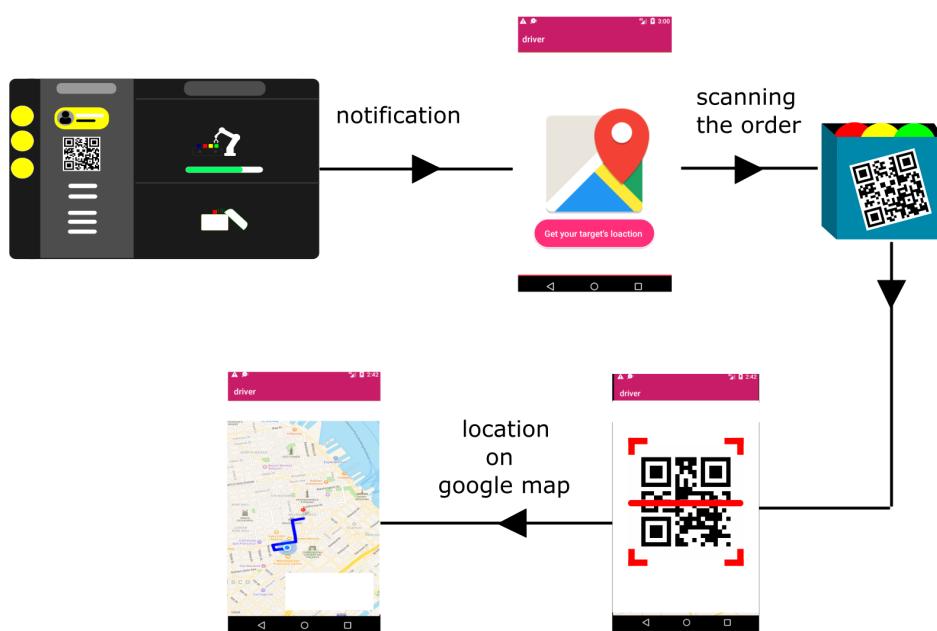


Figure 10.4: Finishing Order Flow

# **Part IV**

# **Integration**

# **Chapter 11**

## **The Factory Integration**

This part is about integrating The Factory inside of The ERP and benefiting from it alongside The BPM.

When trying to integrate any project with our solutions, we will need to reconsider the business processes involved in the business. As stated before the business processes involved

- Manufacturing the products
- Warehousing
- Provide an online market to show the products and sell them
- Delivering the products to the customers

For each business process, we will need to rethink how to migrate it in The ERP and what module in the ERP can be used in it.

### **I Manufacturing**

For Manufacturing the products and controlling the hardware, we will consider using the Manufacturing and IOT modules. With the IOT module, we can control the IOT kit which we will use to connect and control the hardware. And the Manufacturing module which will be used to check for the materials and products to start manufacturing which in the prototype project resembles the sorting process.

### **II Online Market**

The online market is working through the client mobile app to show the products to the user. The module that can be used is the product manager module to show the products and offer the customers a way to buy them.

### **III Warehousing**

The ERP doesn't offer a physical place to be used for warehousing but it has warehousing module to keep track of all the products in the warehouse that can be used to fulfill the need of the warehousing for the Factory.

## **IV Delivering**

The ERP doesn't have any module that can accomplish this mission, but the ERP can be expanded by users to add modules, that don't already exist, to perform the required task (which, in this case, is delivering orders to customers).

## **V Database**

Both The ERP and the Factory have their own database, so before we can completely use The ERP, we will need to migrate the two databases.

We don't offer tools for migrating the database so the users will have to find external tools for database migration, but The BPM offers a way to migrate database by making workflows that will read the tables from one database and writing to the other database but for large databases that will be very hard and time-consuming. The Factory database is small enough for this way to work and migrate the database using it.

## **VI Executing Tasks**

As for executing the business tasks, like making a new order and having it checked by users before proceeding, this will be The BPM task to create a workflow that will wait for the new order and make a user task for the users to evaluate order and then checking the evaluation and deciding to proceed with the order or not.

# **Chapter 12**

## **ERP - BPM - Factory**

This integration will benefit users, as all the services that the ERP modules offer will now be available to the users to use, and are now integrated with the business by the unified database.

Which means users can be benefitting from the CRM module to deal with the customers and help keep in touch with them, the Accounting module to monitor company expenses and customers expenses data, as well as the other modules with the advanced features that they offer.

Users will also benefit from having a unified system that controls the whole business.

These features come from The ERP alone, but when also considering The BPM, users will also have more capability to monitor tasks and get a grasp of the processes run by The BPM to have the ability to track errors and add optimization to their businesses.

The ERP besides The BPM will offer the users a way to automate different processes in an organized and easy way to handle the company growth and help them get rid of system administration hassle by using them on the cloud which will all lead to a managed and organized system with a successful business.

# References

- [1] Object Management Group (OMG). *Web Services Business Process Execution Language Specification, Version 2.0.* (<https://www.omg.org/spec/BPMN/2.0/>). 2011.
- [2] Organization for the Advancement of Structured Information Standards (OASIS). *Web Services Business Process Execution Language (BPEL) Specification, Version 2.0.* (<http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>). 2007.
- [3] Volker Stiehl. *Process-Driven Applications with BPMN*. Springer International Publishing, 2014. ISBN: 978-3-319-07218-0.