**Shipment/delivery company**

Senior Project

by

**Antonio Al Makdissi,**

**Fawzy Ghazawy**

Submitted to the School of Engineering of the

Lebanese International University

Tripoli, Lebanon

in partial fulfillment of the requirements for the degree of

**BACHELOR OF SCIENCE IN COMPUTER ENGINEERING**

**Spring 2022-2023**

**Approved by:**

**Supervisor**  

Dr. Zaher Merhi



**Committee Member**

Dr. TBA



**DEDICATION**

First and foremost, we would like to dedicate this for ourselves, our past selves, to bring to light the progress we have made along the years starting from scratch till what we are today. Though this does not represent our endpoint in life as we are going to strive in order to achieve higher goals via further education and hard work through the many years to come.

**ACKNOWLEDGMENT**

We would like to thank our supervisor and the dean and the rest of the people who spared no effort in guiding and instructing us throughout the different steps of the project. And finally the team members who put a lot of work and dedication to help achieve this project.

**ABSTRACT**

The abstract goes here with double space 12pt font. The abstract consists of one paragraph with first line indentation. The page should be numbered. The abstract goes here with double space 12pt font. The abstract consists of one paragraph with first line indentation. The page should be numbered. The abstract goes here with double space 12pt font. The abstract consists of one paragraph with first line indentation. The page should be numbered. The abstract goes here with double space 12pt font. **The abstract consists of one paragraph with first line indentation**. The page should be numbered. The abstract goes here with double space 12pt font. The abstract consists of one paragraph with first line indentation. The page should be numbered.

**TABLE OF CONTENTS**

[**ACKNOWLEDGMENT iii**](#_heading=h.tyjcwt)

[**ABSTRACT iv**](#_heading=h.3dy6vkm)

[**TABLE OF CONTENTS v**](#_heading=h.1t3h5sf)

[**LIST OF FIGURES viii**](#_heading=h.4d34og8)

[**LIST OF TABLES ix**](#_heading=h.2s8eyo1)

[**LIST OF SYMBOLS x**](#_heading=h.17dp8vu)

[**CHAPTER 1 INTRODUCTION 1**](#_heading=h.3rdcrjn)

[1.1](#_heading=h.26in1rg) Background 1

[1.2](#_heading=h.lnxbz9) Problem Statement 1

[1.3](#_heading=h.35nkun2) General overview of the project 1

[1.4](#_heading=h.1ksv4uv) Thesis Outline 1

[**CHAPTER 2 Survey of Existing Methods and Similar Systems 2**](#_heading=h.44sinio)

[2.1](#_heading=h.2jxsxqh) Introduction 2

[2.2](#_heading=h.z337ya) Method 1 / System 1 2

[2.3](#_heading=h.3j2qqm3) Method 2 / System 2 2

[2.4](#_heading=h.1y810tw) Method 3 / System 3 2

[2.5](#_heading=h.4i7ojhp) Methods/Systems Comparison 2

[2.6](#_heading=h.2bn6wsx) Conclusion and Motivation 3

[**CHAPTER 3 System Design 5**](#_heading=h.qsh70q)

[3.1](#_heading=h.3as4poj) Introduction 5

[3.2](#_heading=h.1pxezwc) Requirements and Specification Analysis 5

[3.2.1](#_heading=h.49x2ik5) Functional Requirements 5

[3.2.2](#_heading=h.2p2csry) Use Case Diagrams 5

[3.3](#_heading=h.147n2zr) System Architecture 6

[3.4](#_heading=h.23ckvvd) Class Diagrams 6

[3.5](#_heading=h.ihv636) Sequence Diagrams 6

[3.6](#_heading=h.32hioqz) Activity Diagrams 7

[3.7](#_heading=h.1hmsyys) Entity-Relationship (ER) Diagrams 7

[3.8](#_heading=h.41mghml) Non-Technical Aspects 7

[3.8.1](#_heading=h.2grqrue) Financial Viability 7

[3.8.2](#_heading=h.3fwokq0) Stakeholders 8

[3.8.3](#_heading=h.1v1yuxt) Scope 8

[3.8.4](#_heading=h.4f1mdlm) Risks 8

[3.8.5](#_heading=h.2u6wntf) Schedule and Milestones 9

[3.8.6](#_heading=h.3tbugp1) Ethical and Social Considerations 9

[3.8.7](#_heading=h.28h4qwu) Environmental and Sustainability Considerations 9

[3.8.8](#_heading=h.nmf14n) Relevant Standards 10

[3.9](#_heading=h.37m2jsg) Conclusion 10

[**CHAPTER 4 Implementation/Simulation and Testing 11**](#_heading=h.1mrcu09)

[4.1](#_heading=h.46r0co2) Introduction 11

[4.2](#_heading=h.2lwamvv) Implementation Tools 11

[4.3](#_heading=h.111kx3o) Implementation Summary 11

[4.4](#_heading=h.3l18frh) Test Cases and Acceptance Criteria 11

[4.5](#_heading=h.206ipza) Conclusion 11

[**CHAPTER 5 Conclusion and Future Work 12**](#_heading=h.4k668n3)

[5.1](#_heading=h.2zbgiuw) Conclusion 12

[5.2](#_heading=h.1egqt2p) Future Work 12

[**APPENDIX A: Implementation Details 13**](#_heading=h.3ygebqi)

[**APPENDIX B: USER Manual 14**](#_heading=h.2dlolyb)

[**APPENDIX C: deployment and configuration Manual 15**](#_heading=h.sqyw64)

[**REFERENCES 16**](#_heading=h.3cqmetx)

**LIST OF FIGURES**

[Figure ‎1‑1: web application preview [1] 3](#_Toc129106660)

[Figure ‎1‑2: paid delivery [1] 4](#_Toc129106661)

**LIST OF TABLES**

Table 2-1: Comparison Table Based on Graphical Interfaces 3

Table 2-2: Comparison Table Based on Content and Functionality 3

Table 2-3: Comparison Table Based on Features 3

**LIST OF SYMBOLS**

MUST LISTED IN ASCENDING ALPHABETICAL ORDER. Examples:

Ω: Resistance

μ: Viscosity

A: Ampere

BW: Bandwidth

CD: Compact Disk

COVID: Corona Virus Disease

ISO: International Standardization Organization

LCD: Liquid Crystal Display

LIU: Lebanese International University

WiFi: Wireless Fidelity

# INTRODUCTION

## Background

Our modern world is full of challenges. That’s why we, as humans, try our best to invent new technologies in order to make our lives less tedious and easier to go by.

We saw the problems that shipment companies face, such as loss of cargo either through piracy or corruption, absence of trust between customer and company due to lack of communication, and insufficient data/intel on the companies’ exchanges.

Hence our attempt at developing our app: a cargo/delivery application that can address several challenges faced by logistics providers and customers, such as lack of visibility, inefficient processes, limited communication, security concerns, competition, and increasing demand. By providing real-time tracking, automating processes, improving communication and transparency, enhancing security, and offering a competitive advantage, a cargo/delivery application can improve customer satisfaction, reduce costs, optimize operations, and provide valuable data insights.



Figure ‎1‑1:Shipyard full of cargo to be delivered [1]

## Problem Statement

Since its foundation, the logistics and transportation industry encounter various difficulties such as inadequate visibility, ineffective procedures, restricted communication, security threats, intense competition, and growing demand for prompt and dependable shipping services. These challenges can cause delivery delays, higher costs, and lower customer satisfaction, which may have negative effects on the profitability and expansion of logistics providers. To solve these challenges, a cargo/delivery application is necessary to deliver instant tracking, automate processes, enhance communication and transparency, boost security, and provide a competitive edge in the market.

* Optimize operations: The application should help logistics providers optimize their operations by reducing manual processes, increasing efficiency, and improving resource allocation.
* Reduce costs: By automating processes and improving efficiency, the application should help logistics providers reduce costs associated with labor, fuel, and other operational expenses.
* Increase revenue: By improving customer satisfaction and providing a competitive advantage, the application should help logistics providers attract new customers and increase revenue.

## General overview of the project

Our project aims to improve the efficiency, transparency, security, and customer satisfaction of logistics and transportation operations. This can be achieved through real-time tracking, process automation, improved communication, enhanced security measures, and a unique value proposition that sets the application apart from competitors.   
The cargo/delivery application aims to provide logistics providers and customers with a comprehensive, user-friendly platform for managing shipments and deliveries, using modern technologies and tools. The application will feature real-time tracking, automated processes, communication tools, enhanced security measures, and data insights to optimize operations and improve customer satisfaction. The real-time tracking feature will leverage GPS and other location-tracking technologies to provide customers with real-time updates on the status and location of their shipments. The process automation feature will automate manual processes involved in managing shipments, such as generating labels, managing inventory, scheduling deliveries, and handling customs documentation. The communication tools feature will provide notifications, alerts, messaging capabilities, and a customer service portal for addressing any issues that may arise. The enhanced security measures feature will include secure login credentials, real-time tracking, and integrated security systems that monitor shipments throughout the delivery process. The data insights feature will provide valuable data on shipment performance, customer behavior, and other key metrics, which can be used to improve operations and make better business decisions.

Figure ‎1‑2: web application preview [1]

## Project Outline

The project aims to develop a cargo/delivery application that will provide real-time tracking, automate processes, improve communication and transparency, enhance security, and offer a competitive advantage to logistics providers. The application will help logistics providers optimize their operations, reduce costs, increase revenue, and improve customer satisfaction. Please follow us throughout the steps of realizing the project, from research to development. As first we are going to conduct a thorough analysis of the requirements and needs of logistics providers and customers to determine the key features and functionalities of the application. We will do this by surveying of existing methods and similar systems in ‎CHAPTER 2. 

Figure ‎1‑3: paid delivery [1]

# Survey of Existing Methods and Similar Systems

## Introduction

The "Requirements gathering and analysis" phase is a critical step in the development of the cargo/delivery application project. In this phase, we will research closely already existing applications (like Wakilni, Aramex and GoShare) to understand the project’s requirements and needs, identify pain points, and define the scope and goals of said project. This phase will involve conducting research, holding meetings, and gathering feedback from logistics providers and customers to develop a comprehensive set of requirements that will guide the development of the application. The project team will use this information to create use cases, user stories, and other documentation that will help ensure that the final product meets the needs of all stakeholders and delivers the desired outcomes. The requirements gathering and analysis phase will be followed by the design and development phases, where the project team will begin building the application based on the requirements identified in this phase.

## System 1: Aramex

<https://www.aramex.com/lb/en/aramex-home?utm_source=google&utm_medium=cpc&utm_campaign=AMX-CONV-WW-SC-IP-20210506&utm_term=general&gclid=CjwKCAiA3pugBhAwEiwAWFzwdTlc3N3GRcFIX8FkntH0epGAOly105NUqFkEGTn-Opi57y01-2TpjRoC5mIQAvD_BwE>

Aramex is a well-known delivery company with their own website that offers:

* e-commerce solutions that assist start-ups and established businesses in supply chain management, technical support, and more [2]
* Logistics Services
* Bundling, Packaging and co-packing
* Aramex Application Programming Interface (APIs)
* Import address book
* Track shipments
* Calculate rates
* Receive shipment updates
* ClickToShip: a separate desktop application without constant link to the internet
* Aramex app: a mobile application that enables Aramex customers worldwide to perform a range of tasks, including tracking deliveries, monitoring shipment progress, managing their accounts, calculating the costs and transit times of upcoming shipments, scheduling pickups and deliveries at preferred locations, dates, and times, making secure payments, and receiving a clear breakdown of delivery dates and custom charges… [2]



Figure ‎2‑1: Aramex's Website [2]

## System 2: Wakilni

<https://wakilni.com/>

Wakilni is a delivery company that offers a range of innovative solutions to help individuals and businesses in the Middle East manage their daily tasks more efficiently. Founded in 2015, the company connects customers with trusted service providers who can take care of tasks like grocery delivery, laundry, home cleaning, and more. Wakilni's personalized approach and use of machine learning algorithms ensure that each customer receives customized service tailored to their unique preferences and behavior.   
Here's what their website offers:

* Donate to children feature.
* Promo code to take a discount.
* Shipment cost around 4$-8$ across Lebanon.
* It takes around 24-72 hours for delivering.
* Join our team feature if they need some workers. Exp: driver.
* Live chat feature: they will ask for your name and the email
* A marketplace page that shows all the businesses in Lebanon that are registered in Wakelin and they provide the customer with categories to find what he is searching for; and provide a link for each one of them.
* Extra mile fund donations: As of October 2020, Wakilni will donate 750 liras from each commissioned delivery to its Extra Mile Fund*.* This money will go towards helping members of the Wakilni community reboot their businesses by accessing needed services, whether in terms of online exposure, marketing, legal counseling, etc. and Wakilni will be sponsoring this access to services. [3]



Figure ‎2‑2: Wakilni Website [3]

## System 3: GoShare

<https://goshare.co/>

GoShare is a rapidly growing logistics and transportation company based in San Diego, California. Founded in 2014, the company has quickly become a leading provider of on-demand delivery and moving services in the United States.  
We personally chose GoShare as it has received numerous accolades for its innovative approach to logistics and transportation. The company was named one of the "Most Promising Startups" at the 2017 Forbes Under 30 Summit, and it has been featured in publications like the Wall Street Journal, TechCrunch, and CNBC.   
Here are some key points about GoShare:

* GoShare offers a variety of services, including local delivery, hauling, and moving. Customers can use the company's app or website to book a driver and a truck for a range of jobs, from small-item delivery to full-scale residential moves.
* One of GoShare's key selling points is its on-demand availability. The company's large network of independent contractors allows it to offer fast service, with many jobs completed within an hour of booking.
* GoShare drivers are rigorously vetted and insured, so customers can trust that their items are in safe hands. Drivers are also equipped with a range of tools and materials to help them complete jobs efficiently and effectively.
* GoShare is committed to sustainability and reducing its carbon footprint. The company's fleet of trucks includes electric and hybrid vehicles, and it encourages its drivers to adopt eco-friendly driving habits.
* In addition to its core services, GoShare also offers business solutions for companies looking to streamline their logistics operations. The company's platform allows businesses to manage deliveries, track shipments, and access real-time data on their delivery performance.

Overall, GoShare is a dynamic and rapidly expanding company that is shaking up the traditional logistics industry. With its on-demand availability, focus on sustainability, and commitment to innovation, GoShare is poised for continued success in the years to come. [4]



Figure ‎2‑3: GoShare [4]

## Methods/Systems Comparison

This paragraph in meant to highlight the advantages and shortcomings of each of the previously stated methods with respect to certain criteria.

**Table 2-1: Comparison Table Based on Graphical Interfaces**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criterion 1** | **Aramex** | **Wakilni** | **GoShare** |
| **Graphical Interface** |
| Good user interface | **🗸** | **🗸** | **🗸** |
| Easy and effective navigation | **🗴** | **🗸** | **🗴** |
| Simple and professional Design | **🗸** | **🗸** | **🗸** |
| Responsive | **🗴** | **🗴** | **🗸** |

A well-designed graphical user interface (GUI) is essential for the success of an application. A good user interface (UI), easy navigation, professional design, and responsiveness are crucial components. These features enhance the user experience and increase usage and satisfaction. A well-designed GUI with a good UI, easy navigation, professional design, and responsiveness improves the usability and appeal of an application.

**Table 2-2: Comparison Table Based on Content and Functionality**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criterion 2** | **Aramex** | **Wakilni** | **GoShare** |
| **Content and Functionality** |
| Quality content structure | **🗸** | **🗸** | **🗸** |
| Usability | **🗴** | **🗸** | **🗸** |
| Dynamic content | **🗴** | **🗸** | **🗸** |
| Content management system | **🗴** | **🗴** | **🗸** |

Quality content structure, usability, dynamic content, and content management systems are all essential components of a successful application. Quality content structure ensures that the information within the application is presented in a clear and logical manner, making it easy for users to find what they need. Usability refers to the ease with which users can navigate the application, complete tasks, and achieve their goals. Dynamic content, such as personalized recommendations or real-time updates, can enhance user engagement and satisfaction. Finally, a content management system (CMS) allows for efficient and effective management of the application's content, ensuring that it remains up-to-date and relevant. Together, these features help create an application that is intuitive, engaging, and valuable to its users.

**Table 2-3: Comparison Table Based on Features**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criterion 3** | **Aramex** | **Wakilni** | **GoShare** |
| **Features** |
| Security measures | **🗸** | **🗴** | **🗸** |
| Third party integration | **🗴** | **🗸** | **🗸** |
| Accessible content and location | **🗸** | **🗸** | **🗸** |
| Registration form | **🗸** | **🗴** | **🗸** |

For any application, security, third-party integration, accessibility, and registration forms are important considerations. Security measures protect user data, while third-party integrations enhance functionality. Accessible content ensures all users can access the application, and registration forms personalize the user experience. Taking these factors into account leads to a secure, functional, accessible, and personalized application, increasing user satisfaction and engagement.

## Conclusion and Motivation

Through the study of other systems, we have identified the essential components that a successful application should possess. These components include a well-designed graphical user interface (GUI), quality content structure, usability, dynamic content, content management systems, security measures, third-party integration, accessible content and location, and registration forms. A well-designed GUI with a good UI, easy navigation, professional design, and responsiveness enhances the user experience and increases usage and satisfaction. Quality content structure ensures that information is presented logically, while usability and dynamic content enhance user engagement. A content management system (CMS) allows for efficient management of the application's content. Security measures, third-party integration, accessible content and location, and registration forms are also important considerations. By taking these factors into account, we can create an application that is secure, functional, accessible, and personalized, leading to increased user satisfaction and engagement.

# System Design

## Introduction

This paragraph in meant to introduce the topics to be covered in this chapter.

## Requirements and Specification Analysis

This section is meant to explain to the reader use cases for the project, activity diagrams, as well as specification, documentation and requirements of the project. This section can have sub-sections.

### Functional Requirements

Here you have to list the operations and activities that a system must be able to perform.

### Use Case Diagrams

A use case diagram is the primary form of system/software requirements for a new software program. A key concept of use case modeling is that it helps us design a system from the end user's perspective. A use case diagram is usually simple:

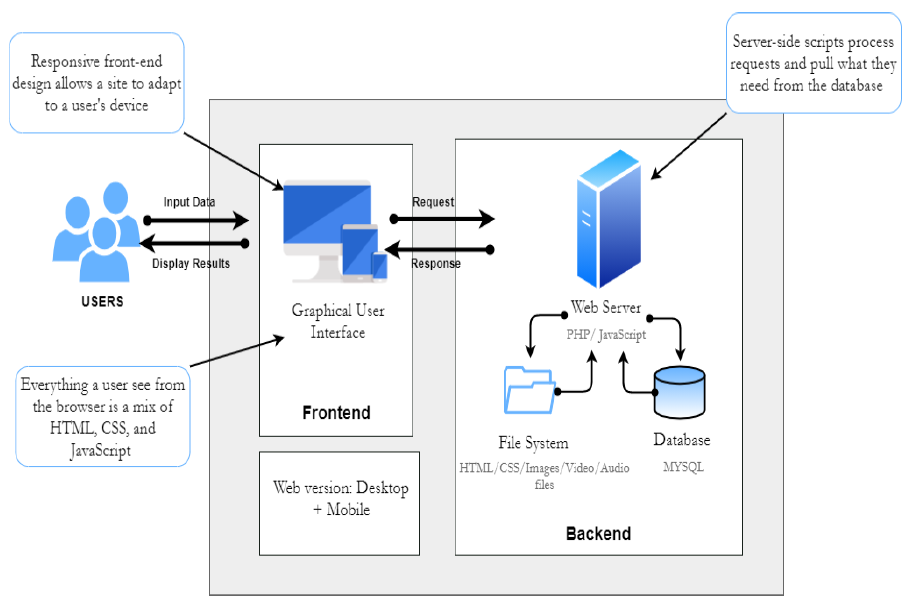
* It only summarizes **some of the relationships** between use cases, actors, and systems.
* It does **not show the order** in which steps are performed to achieve the goals of each use case.

Check this tutorial:

<https://www.youtube.com/watch?v=zid-MVo7M-E>

## System Architecture

Define the overall architecture of the system. Layered, Dataflow, Multi-tier, Client-Server, etc. It is worth noting the underlying technologies to be used. Draw a figure that shows the complete system Example of such figures:



**Figure ‎3-1- System Architecture**

## Class Diagrams

The static design view of the system. It includes all the underlying classes designed and their hierarchy. Check this tutorial: <https://www.youtube.com/watch?v=UI6lqHOVHic>

## Sequence Diagrams

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

Check this tutorial: <https://www.youtube.com/watch?v=pCK6prSq8aw>

## Activity Diagrams

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination.

Check this tutorial: <https://www.youtube.com/watch?v=3Hw_VXea73o>

## Entity-Relationship (ER) Diagrams

An entity relationship diagram (ERD) is a graphical representation of entities and their relationships to each other, typically used for modeling the organization of data within databases or information systems. Check these tutorials starting with the first one:

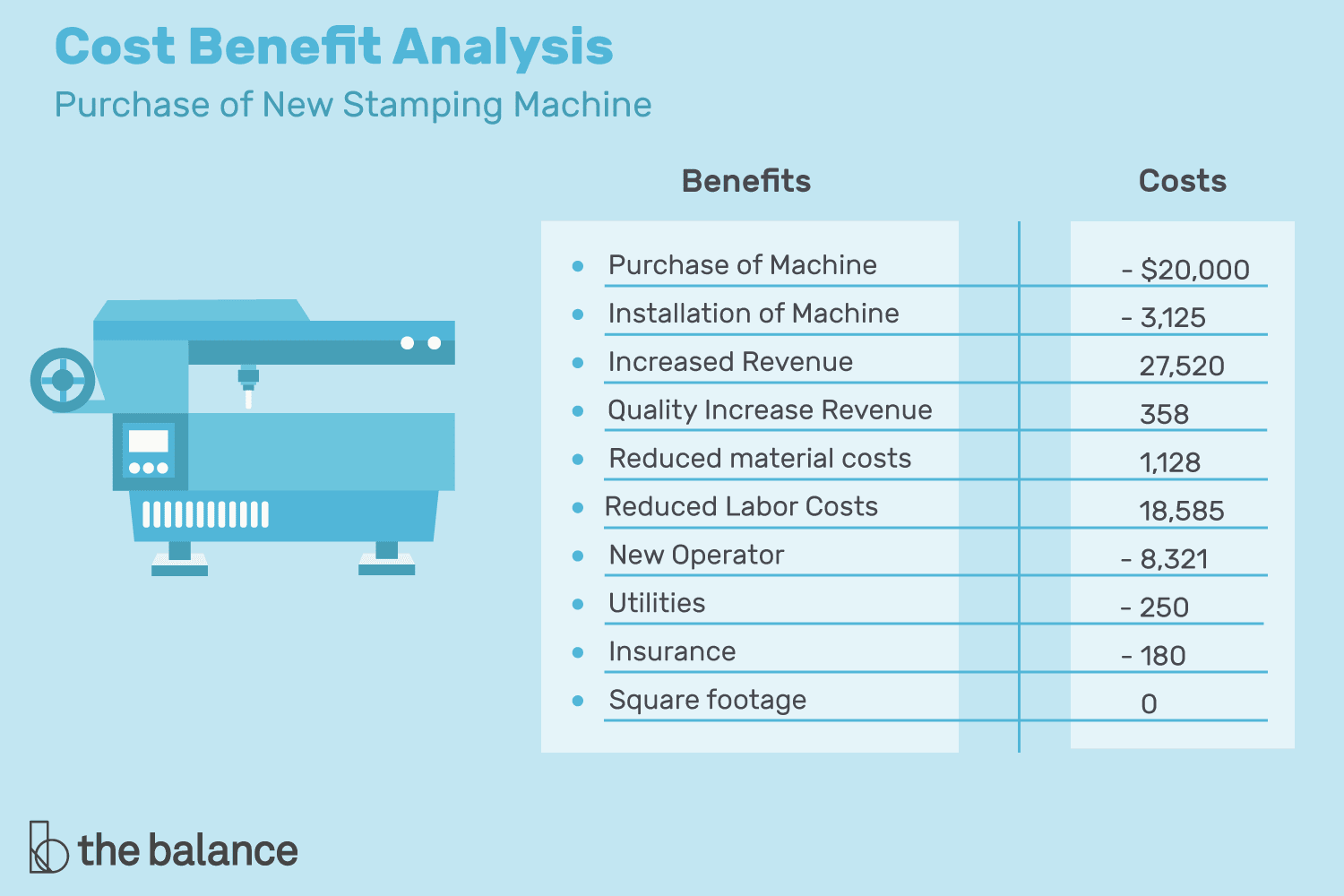
Tutorial 1: <https://www.youtube.com/watch?v=QpdhBUYk7Kk>

Tutorial 2: <https://www.youtube.com/watch?v=-CuY5ADwn24>

## Non-Technical Aspects

### Financial Viability

A cost benefit analysis of the project. Here is an example:



**Figure ‎3-2- Cost Benefit Analysis**

### Stakeholders

Who will benefit? Who may be harmed? Who should have a say in how the project works?

### Scope

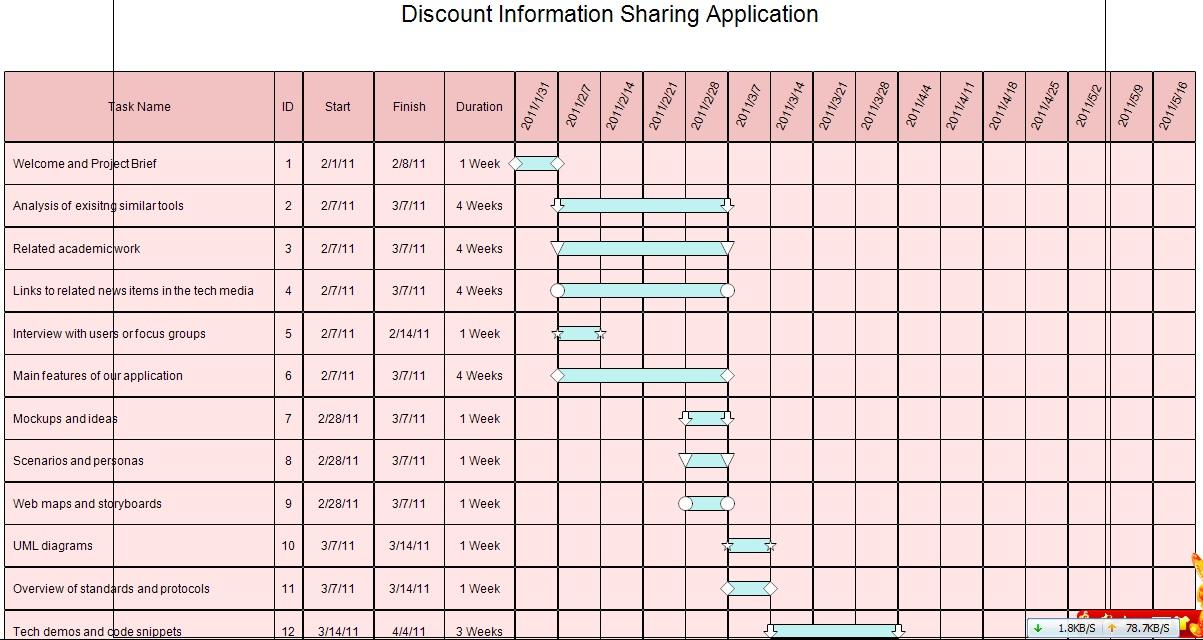
Exactly what will be done in the project, sometimes it is helpful to state what will not be done in the project.

### Risks

Things that may stop the project from achieving the goals in the scope.

### Schedule and Milestones

What will be completed and when. Here is an example:



**Figure ‎3-3- Scheduling Tasks and Milestones**

### Ethical and Social Considerations

Are there any ethical issues that should be taken into consideration when completing the design?

### Environmental and Sustainability Considerations

Are there any environmental issues to write about? It is ok to write that this project has no environmental impact.

### Relevant Standards

List the technical (and possibly the non-technical) standards that are relevant to your design. Examples are the WiFi standard (IEEE) and International Standardization Organization (ISO).

## Conclusion

This paragraph in meant to draw conclusions highlighting the main ideas in this chapter.

# Implementation/Simulation and Testing

## Introduction

This paragraph in meant to introduce the topics to be covered in this chapter.

## Implementation Tools

List the tools used for implementing the system. This includes hardware, compilers, IDE, frameworks, CASE tools, etc...

## Implementation Summary

Description of detailed implementation steps. Demonstrate the typical code fragments (details of implementation, e.g. source code listings must be included in an appendix and saved on an accompanying CD/DVD)

## Test Cases and Acceptance Criteria

Describe the test cases used and the acceptance criteria.

## Conclusion

This paragraph in meant to draw conclusions highlighting the main ideas in this chapter.

# Conclusion and Future Work

## Conclusion

The cargo/delivery application project aims to develop a solution that addresses the challenges faced by logistics providers and customers, such as lack of visibility, inefficient processes, limited communication, security concerns, competition, and increasing demand. By providing real-time tracking, process automation, improved communication, enhanced security measures, a unique value proposition, and valuable data insights, the cargo/delivery application will help logistics providers optimize their operations, reduce costs, increase revenue, and improve customer satisfaction.

## Future Work

Describe the opportunities for expanding the work done in this thesis.

**APPENDIX A:   
Implementation Details**

Any details not fit in chapter 5: e.g. detailed calculation, complex algorithms, etc…

**APPENDIXB:  
 USER Manual**

Fill in the instruction manual for using the application

**APPENDIXC:   
deployment and configuration Manual**

Outline the deployment and configuration details in addition to any know troubleshooting techniques.

**REFERENCES**

|  |  |
| --- | --- |
| [1] | [Online]. Available: https://pixabay.com/images/search/shipment/. |