

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**SYSTEM REQUIREMENTS SPECIFICATION  
CSE 4317: SENIOR DESIGN II  
SUMMER 2021**



**LAMINAR WORKFLOW  
QUICK QUOTE**

**SINDHU PARAJULI  
LAITH MARZOUQ  
ALISHA KUNWAR  
NISHAN THAPA  
ERIC KENG**

## REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	03.19.2021	EK, LM, SP, NT, AK	document creation
0.2	03.20.2021	EK	Sections: 1, 3.0, 3.2, 3.10, 8.1, 8.2, 9
0.3	03.20.2021	AK, NT	Sections: 3.3, 3.5, 3.6, 3.8, 4, 4.1, 8.3, 8.4
0.4	03.20.2021	SP	Sections: 3.4, 3.7, 4.2, 7, 7.1
0.5	03.20.2021	LM	Sections: 2, 3.9, 5, 7.2
1.0	03.22.2021	EK	Final Draft Review
2.0	05.04.2021	SP, AK, LM	Version-2 SRS Review
2.1	08.03.2021	SP, AK, NT, LM, AK	Close Out Review

# CONTENTS

<b>1</b>	<b>Product Concept</b>	<b>8</b>
1.1	Purpose and Use . . . . .	8
1.2	Intended Audience . . . . .	8
<b>2</b>	<b>Product Description</b>	<b>9</b>
2.1	Features & Functions . . . . .	9
2.2	External Inputs & Outputs . . . . .	9
2.3	Product Interfaces . . . . .	9
<b>3</b>	<b>Customer Requirements</b>	<b>11</b>
3.1	Customizable Landing Page (CUSTOMER) . . . . .	11
3.1.1	Description . . . . .	11
3.1.2	Source . . . . .	11
3.1.3	Constraints . . . . .	11
3.1.4	Standards . . . . .	11
3.1.5	Priority . . . . .	11
3.2	Parsing of DXF File (System) . . . . .	11
3.2.1	Description . . . . .	11
3.2.2	Source . . . . .	11
3.2.3	Constraints . . . . .	12
3.2.4	Standards . . . . .	12
3.2.5	Priority . . . . .	12
3.3	Individual Account Sign Up and Login . . . . .	12
3.3.1	Description . . . . .	12
3.3.2	Source . . . . .	12
3.3.3	Constraints . . . . .	12
3.3.4	Standards . . . . .	12
3.3.5	Priority . . . . .	12
3.4	Modifiable Quoting Parameters (USER) . . . . .	12
3.4.1	Description . . . . .	12
3.4.2	Source . . . . .	12
3.4.3	Constraints . . . . .	12
3.4.4	Standards . . . . .	12
3.4.5	Priority . . . . .	12
3.5	Modifiable Quoting Parameters (CUSTOMER) . . . . .	13
3.5.1	Description . . . . .	13
3.5.2	Source . . . . .	13
3.5.3	Constraints . . . . .	13
3.5.4	Standards . . . . .	13
3.5.5	Priority . . . . .	13
3.6	Get price quote (USER) . . . . .	13
3.6.1	Description . . . . .	13
3.6.2	Source . . . . .	13
3.6.3	Constraints . . . . .	13
3.6.4	Standards . . . . .	13
3.6.5	Priority . . . . .	13

3.7	Submit and Pay for Order After Receiving Quote (USER)	13
3.7.1	Description	13
3.7.2	Source	13
3.7.3	Constraints	13
3.7.4	Standards	14
3.7.5	Priority	14
3.8	Drag and Drop File Feature	14
3.8.1	Description	14
3.8.2	Source	14
3.8.3	Constraints	14
3.8.4	Standards	14
3.8.5	Priority	14
3.9	Track Order Status	14
3.9.1	Description	14
3.9.2	Source	14
3.9.3	Constraints	14
3.9.4	Standards	14
3.9.5	Priority	14
3.10	Track Order History	14
3.10.1	Description	14
3.10.2	Source	15
3.10.3	Constraints	15
3.10.4	Standards	15
3.10.5	Priority	15
3.11	Quoting Based on Other File Types	15
3.11.1	Description	15
3.11.2	Source	15
3.11.3	Constraints	15
3.11.4	Standards	15
3.11.5	Priority	15
<b>4</b>	<b>Packaging Requirements</b>	<b>16</b>
4.1	Application Accessible Through Web Browser	16
4.1.1	Description	16
4.1.2	Source	16
4.1.3	Constraints	16
4.1.4	Standards	16
4.1.5	Priority	16
4.2	Compatible With Smartphones	16
4.2.1	Description	16
4.2.2	Source	16
4.2.3	Constraints	16
4.2.4	Standards	16
4.2.5	Priority	16

<b>5</b>	<b>Performance Requirements</b>	<b>17</b>
5.1	Instant price quote . . . . .	17
5.1.1	Description . . . . .	17
5.1.2	Source . . . . .	17
5.1.3	Constraints . . . . .	17
5.1.4	Standards . . . . .	17
5.1.5	Priority . . . . .	17
5.2	Consistent Output Generation . . . . .	17
5.2.1	Description . . . . .	17
5.2.2	Source . . . . .	17
5.2.3	Constraints . . . . .	17
5.2.4	Standards . . . . .	17
5.2.5	Priority . . . . .	17
<b>6</b>	<b>Safety Requirements</b>	<b>18</b>
6.1	Data Confidentiality and Safety . . . . .	18
6.1.1	Description . . . . .	18
6.1.2	Source . . . . .	18
6.1.3	Constraints . . . . .	18
6.1.4	Standards . . . . .	18
6.1.5	Priority . . . . .	18
6.2	Laboratory equipment lockout/tagout (LOTO) procedures . . . . .	18
6.2.1	Description . . . . .	18
6.2.2	Source . . . . .	18
6.2.3	Constraints . . . . .	18
6.2.4	Standards . . . . .	18
6.2.5	Priority . . . . .	18
<b>7</b>	<b>Maintenance &amp; Support Requirements</b>	<b>19</b>
7.1	Fixing Bugs encountered after release and customer feedback . . . . .	19
7.1.1	Description . . . . .	19
7.1.2	Source . . . . .	19
7.1.3	Constraints . . . . .	19
7.1.4	Standards . . . . .	19
7.1.5	Priority . . . . .	19
7.2	All Code Saved on GitHub . . . . .	19
7.2.1	Description . . . . .	19
7.2.2	Source . . . . .	19
7.2.3	Constraints . . . . .	19
7.2.4	Standards . . . . .	19
7.2.5	Priority . . . . .	19
<b>8</b>	<b>Other Requirements</b>	<b>20</b>
8.1	React JS for Frontend . . . . .	20
8.1.1	Description . . . . .	20
8.1.2	Source . . . . .	20
8.1.3	Constraints . . . . .	20
8.1.4	Standards . . . . .	20

8.1.5	Priority . . . . .	20
8.2	Spring Boot for Backend API . . . . .	20
8.2.1	Description . . . . .	20
8.2.2	Source . . . . .	20
8.2.3	Constraints . . . . .	20
8.2.4	Standards . . . . .	20
8.2.5	Priority . . . . .	20
8.3	Database Server . . . . .	21
8.3.1	Description . . . . .	21
8.3.2	Source . . . . .	21
8.3.3	Constraints . . . . .	21
8.3.4	Standards . . . . .	21
8.3.5	Priority . . . . .	21
8.4	Full Stack Application . . . . .	21
8.4.1	Description . . . . .	21
8.4.2	Source . . . . .	21
8.4.3	Constraints . . . . .	21
8.4.4	Standards . . . . .	21
8.4.5	Priority . . . . .	21
<b>9</b>	<b>Future Items</b>	<b>22</b>
9.1	Quoting Based on Other File Types . . . . .	22
9.1.1	Description . . . . .	22
9.1.2	Source . . . . .	22
9.1.3	Constraints . . . . .	22
9.1.4	Standards . . . . .	22
9.1.5	Priority . . . . .	22

## LIST OF FIGURES

1	Quick-Quote conceptual drawing . . . . .	8
2	External Input & Output . . . . .	9
3	Produced Interface . . . . .	10

# 1 PRODUCT CONCEPT

The product proposition is a Quick-Quote application for 2D cutting services such as laser, plasma, and water-jet cutting. The intent is to create an application that companies, who offer cutting services, can use to offer instant-quotes for their services. Customers will be able to customize their landing page to represent their company brand in a way that presents the instant-quoting service as their own. Users will be able to drag and drop DXF files into the Quick-Quote application and receive an instant price for their desired product.

## 1.1 PURPOSE AND USE

The purpose of our product is give companies access to instant-quoting services to keep up with the growing market demand for instant online shopping. Instant quoting services exist but are normally provided by proprietary software. Our goal is to offer a subscription based product that makes instant-quoting software available to all companies.

## 1.2 INTENDED AUDIENCE

The primary focus is to create a product that will benefit companies who offer services based off of 2D models for cutting. Our primary customer audience is companies who offer quoting services to their customers using a 2D model in a DXF file format typically for metal cutting.

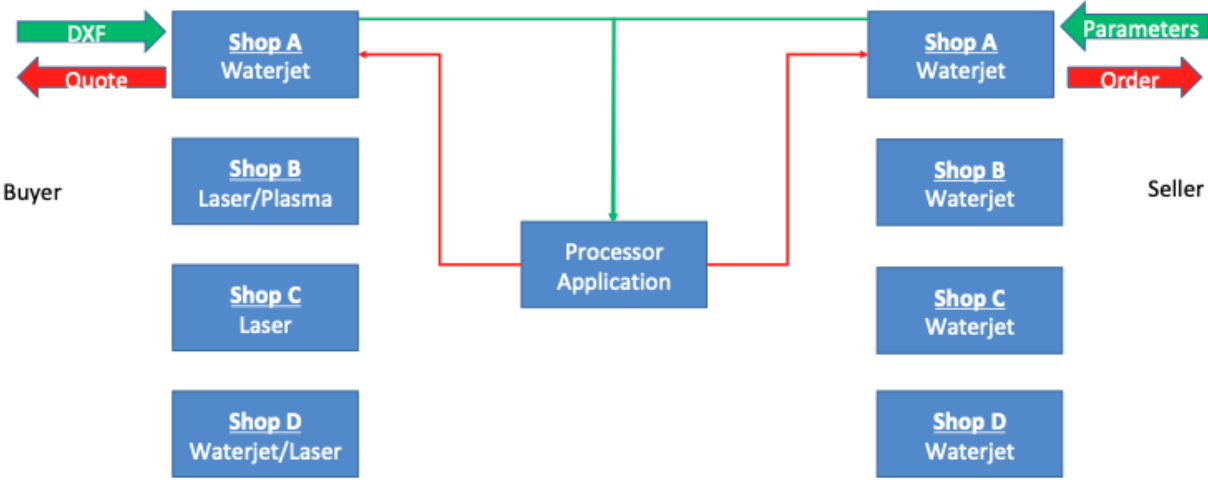


Figure 1: Quick-Quote conceptual drawing



## 2 PRODUCT DESCRIPTION

This section provides the reader with an overview of Quick Quote. The primary operational aspects of the product, from the perspective of end users, maintainers and administrators, are defined here. The key features and functions found in the product, as well as critical user interactions and user interfaces are described in detail.

### 2.1 FEATURES & FUNCTIONS

This application will take a DXF file, input from Users and customers, and output a quote within minutes for the buyer to accept or decline. If the quote is accepted and payment is completed, then an order confirmation will be sent to the buyer and seller. Job shops will also be able to customize their instant quote landing page as they see fit.

This application will not take a 3D image, or files other than DXF.

### 2.2 EXTERNAL INPUTS & OUTPUTS

Name	Input/Output	Description	Use
Users Personal Information	Input	Users will provide their first/last name, address, and phone number	Used to contact user and ship final produce
Users Cutting Information	Input	Users will choose the material, cutting process, and thickness	Used to calculate the price of finished produced
Users Payment Information	Input	Users will provide their credit card information	Used for payment
User DXF file	Input	User will drag and drop DXF for 2D image	Used by parsing algorithm
Job Shop Parameters	Input	Job shops will provide \$/in, setup fee, material cost, shipping cost	Used to calculate the price of finished produced
Instant quote generation	Output	Algorithm will use users cutting information, DXF, and job shops parameters to generate a quote	Used by customers to accept or decline job pricing
Order confirmation	Output	Once quote is accepted and payment is received, order confirmation is sent	Used by job shops to place order

Figure 2: External Input & Output

### 2.3 PRODUCT INTERFACES

As you can see in figure 3, the customer will send the DXF file and the cutting/material information to the parsing algorithm. The algorithm will parse the DXF file and output the total length of cut, number of pierce points, and total area/volume of the material. These values and the shops custom pricing and parameters will be used to calculate the quote, which will be sent to the customer in a matter of minutes. If the customer likes the price and wants to place the order, they will be asked for their credit card information and proceed to payment. The order is now finalized and an order confirmation will be sent to the user and shop, finally the shops will cut and ship the product to the customer.

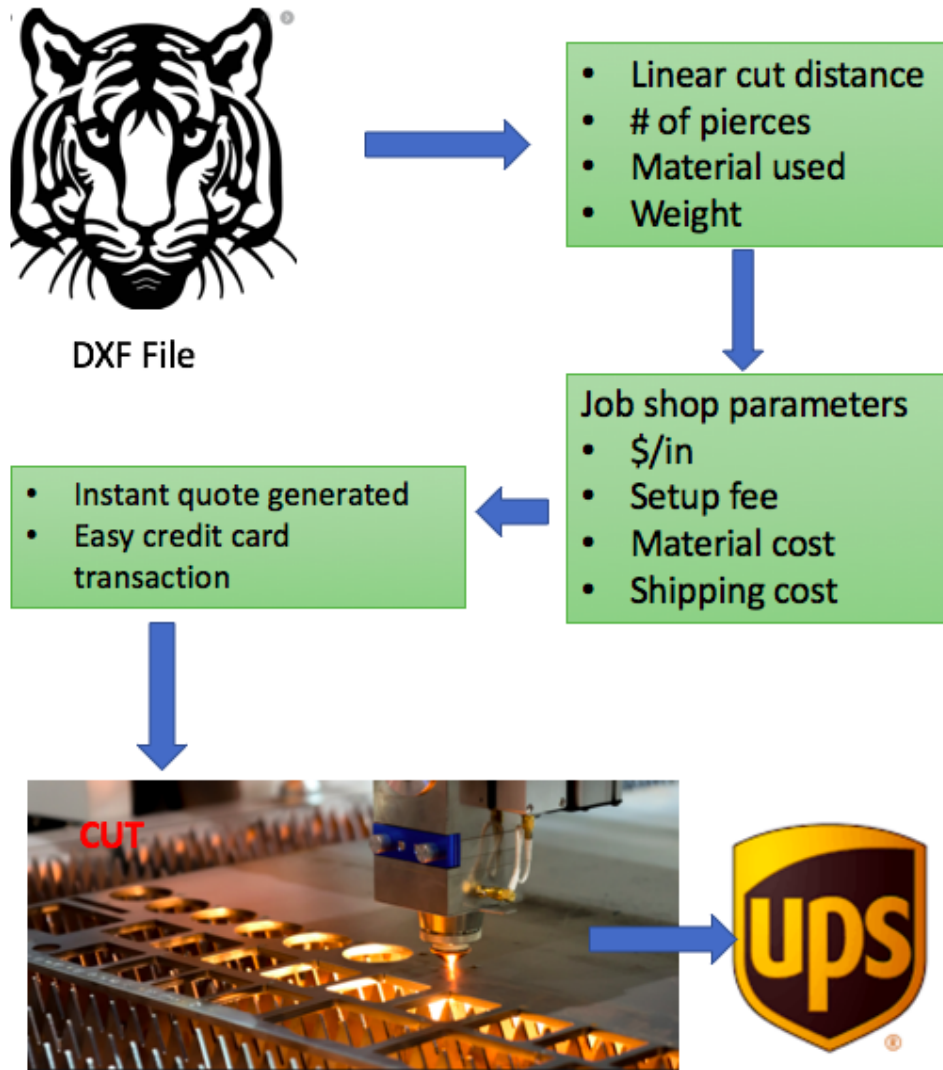


Figure 3: Produced Interface

### 3 CUSTOMER REQUIREMENTS

Quick-Quote is a web application that provides an instant quoting services for 2D cutting companies. Companies will use this service to offer instant quotes to users (their buyers).

- Sponsor of this product is Robert Gullette
- Customers are 2D cutting companies
- Users are the buyers who request quotes for 2D cutting

#### 3.1 CUSTOMIZABLE LANDING PAGE (CUSTOMER)

##### 3.1.1 DESCRIPTION

A customizable landing page is the primary marketable aspect of Quick-Quote. The Quick-Quote application needs to have a formattable landing page for companies to use to represent their company.

- Reserved locations at the top and side for companies to upload their picture/logo.
- General information section which allows companies to put their contact information
- Menu bar and background colors will be customizable.
- General formatting locations will remain the same for all customers
- Landing page will have a custom address such as: [quickquote.com/johndocuttingservice](http://quickquote.com/johndocuttingservice).

##### 3.1.2 SOURCE

Sponsor

##### 3.1.3 CONSTRAINTS

Too many custom formatting options will increase the complexity of the application and potentially harm the usability of the instant quoting service

##### 3.1.4 STANDARDS

N/A

##### 3.1.5 PRIORITY

High

#### 3.2 PARSING OF DXF FILE (SYSTEM)

##### 3.2.1 DESCRIPTION

The system will be able to parse the DXF files uploaded by the user. The system should extract data like total linear distance, number of pierces, area of material used, max length, width and nesting pattern of parts from the DXF file.

##### 3.2.2 SOURCE

Quick-Quote Developers

### **3.2.3 CONSTRAINTS**

The file must be in DXF format.

### **3.2.4 STANDARDS**

N/A

### **3.2.5 PRIORITY**

Critical

## **3.3 INDIVIDUAL ACCOUNT SIGN UP AND LOGIN**

### **3.3.1 DESCRIPTION**

The system will be able to support sign up and login for both 'customers' and 'users' of the application. As the underlying interface (features) differs from individual account to customer account, different logins will be supported by the system.

### **3.3.2 SOURCE**

Quick-Quote Developers

### **3.3.3 CONSTRAINTS**

Might confuse both users and customers while trying to login.

### **3.3.4 STANDARDS**

N/A

### **3.3.5 PRIORITY**

High

## **3.4 MODIFIABLE QUOTING PARAMETERS (USER)**

### **3.4.1 DESCRIPTION**

Users should be able to enter quantities and thickness per DXF files, select materials per DXF files. User should be able to enter shipping address and desired lead time.

### **3.4.2 SOURCE**

Sponsor

### **3.4.3 CONSTRAINTS**

The application should have buyer side interface.

### **3.4.4 STANDARDS**

N/A

### **3.4.5 PRIORITY**

Critical

### **3.5 MODIFIABLE QUOTING PARAMETERS (CUSTOMER)**

#### **3.5.1 DESCRIPTION**

The shop should be able to set up their specific price parameters, and manage buyer side templates.

#### **3.5.2 SOURCE**

Sponsor

#### **3.5.3 CONSTRAINTS**

The application should have shop side (material setup) interface.

#### **3.5.4 STANDARDS**

N/A

#### **3.5.5 PRIORITY**

Critical

### **3.6 GET PRICE QUOTE (USER)**

#### **3.6.1 DESCRIPTION**

The user should get total price estimation for their order. The system will parse the design data from the DXF file. Then the system should calculate the total price for the order based on the price parameters added by Customer (Job shops). The system should show the total price to the users.

#### **3.6.2 SOURCE**

Sponsor

#### **3.6.3 CONSTRAINTS**

The customer (job shops) should enter specific price parameters into the system.

#### **3.6.4 STANDARDS**

N/A

#### **3.6.5 PRIORITY**

Critical

### **3.7 SUBMIT AND PAY FOR ORDER AFTER RECEIVING QUOTE (USER)**

#### **3.7.1 DESCRIPTION**

The user should be able to pay and place their order. Once user gets the price quote of their order, they can choose to pay for the the order and place the order. The payment method used will be mainly through debit card, credit card or through PayPal.

#### **3.7.2 SOURCE**

Quick-Quote Developers

#### **3.7.3 CONSTRAINTS**

Payment must be approved to place the order. DXF parsing must be successful.

### **3.7.4 STANDARDS**

N/A

### **3.7.5 PRIORITY**

Critical

## **3.8 DRAG AND DROP FILE FEATURE**

### **3.8.1 DESCRIPTION**

The user should be able to drag and drop the files in the application interface to get their quotes. The drag and drop interaction should be compatible including the 'upload' option for the ease of the users to get instant quoting by the application.

### **3.8.2 SOURCE**

Sponsor

### **3.8.3 CONSTRAINTS**

Drag-Drop action for many files can be tedious for the users when compared to standard 'upload' option.

### **3.8.4 STANDARDS**

N/A

### **3.8.5 PRIORITY**

High

## **3.9 TRACK ORDER STATUS**

### **3.9.1 DESCRIPTION**

Users should be able to view and track their order status. The job shops should be able to track and manage user metrics.

### **3.9.2 SOURCE**

Quick-Quote Developers

### **3.9.3 CONSTRAINTS**

The system should update the status of the order received.

### **3.9.4 STANDARDS**

N/A

### **3.9.5 PRIORITY**

Moderate

## **3.10 TRACK ORDER HISTORY**

### **3.10.1 DESCRIPTION**

Users should be able to view all orders made on their account, and Customers (job shops) should be able to view all orders made to their shop

### **3.10.2 SOURCE**

Quick-Quote Developers

### **3.10.3 CONSTRAINTS**

An account setup with a login must be created. Past orders have to be made.

### **3.10.4 STANDARDS**

N/A

### **3.10.5 PRIORITY**

Moderate

## **3.11 QUOTING BASED ON OTHER FILE TYPES**

### **3.11.1 DESCRIPTION**

The goal of Quick-Quote is to create a product that allows companies to offer a competitive instant quote service. Offering quoting based off several file types will increase the market appeal for this product.

### **3.11.2 SOURCE**

Quick-Quote Developers

### **3.11.3 CONSTRAINTS**

Some file types to be considered:

- STEP (.stp, .step)
- SOLIDWORKS (.sldprt)
- STL (.stl)
- Parasolid
- Autodesk Inventor (.ipt)

### **3.11.4 STANDARDS**

Market trends show these are common files types supported for instant-quote services for 2D cutting

### **3.11.5 PRIORITY**

Future

## **4 PACKAGING REQUIREMENTS**

It is a software project so it does not have any hardware component. It is a cross-platform web-based application.

### **4.1 APPLICATION ACCESSIBLE THROUGH WEB BROWSER**

#### **4.1.1 DESCRIPTION**

The final product will be delivered to the users through the public domain on the web. It should be compatible with all kinds of available web-browsers. The application will run on a cloud based server.

#### **4.1.2 SOURCE**

Quick-Quote Developers

#### **4.1.3 CONSTRAINTS**

User should have access to the internet.

#### **4.1.4 STANDARDS**

The standard for the final product is to make sure the web application is user-friendly, simple, and intuitive.

#### **4.1.5 PRIORITY**

High

### **4.2 COMPATIBLE WITH SMARTPHONES**

#### **4.2.1 DESCRIPTION**

The product will be compatible to be accessed on smart-phones. The application will be responsive so that users won't have problems accessing it via their mobile devices.

#### **4.2.2 SOURCE**

Team

#### **4.2.3 CONSTRAINTS**

User should have access to the internet. Website frontend view needs to adjust to smartphone screens.

#### **4.2.4 STANDARDS**

N/A

#### **4.2.5 PRIORITY**

High



## **5 PERFORMANCE REQUIREMENTS**

Quick Quote needs to be efficient and consistent. Once the user has completed filling in their information and uploaded their DXF file, their quote must be generated within minutes. Therefore, performance requirements play an important role for this application

### **5.1 INSTANT PRICE QUOTE**

#### **5.1.1 DESCRIPTION**

DXF file parsing and price calculation should not take more than a few seconds.

#### **5.1.2 SOURCE**

Sponsor

#### **5.1.3 CONSTRAINTS**

3 seconds max time. Standard internet speeds used for testing. 10Mbps upload speed.

#### **5.1.4 STANDARDS**

N/A

#### **5.1.5 PRIORITY**

Critical

### **5.2 CONSISTENT OUTPUT GENERATION**

#### **5.2.1 DESCRIPTION**

Application output should remain consistent and reliable when used on separate occasions. (Example: if I use the same DXF file with the same parameters, I should always get the same generated price)

#### **5.2.2 SOURCE**

Team

#### **5.2.3 CONSTRAINTS**

N/A

#### **5.2.4 STANDARDS**

N/A

#### **5.2.5 PRIORITY**

Critical

## **6 SAFETY REQUIREMENTS**

Quick Quote is a web-based application for providing price quote on metal cutting to the users. The application will be virtual and developed online. There is not any safety requirements users should follow to use the app. However, the major concern of the app will be the safety of customer data and information.

### **6.1 DATA CONFIDENTIALITY AND SAFETY**

#### **6.1.1 DESCRIPTION**

The system should store the critical user information securely. During sign up or payment process, customer will provide critical information like name, address, credit card information etc. The system should follow the most secure policy for the proper confidentiality of those information.

#### **6.1.2 SOURCE**

Quick-Quote Developers

#### **6.1.3 CONSTRAINTS**

The system should not store payment information to the database.

#### **6.1.4 STANDARDS**

The most acceptable standards for personal information will be used.

#### **6.1.5 PRIORITY**

Critical

### **6.2 LABORATORY EQUIPMENT LOCKOUT/TAGOUT (LOTO) PROCEDURES**

#### **6.2.1 DESCRIPTION**

Our team will not require any laboratory equipment for the development of this product. In case of emergency, team might borrow personal laptops, computers, or other accessories from the UTA library. Any such devices used in the development of the project will be used in accordance with OSHA standard LOTO procedures.

#### **6.2.2 SOURCE**

CSE Senior Design laboratory policy

#### **6.2.3 CONSTRAINTS**

Equipment usage, due to lock removal policies, will be limited to availability of the course instructor and designated teaching assistants.

#### **6.2.4 STANDARDS**

Occupational Safety and Health Standards 1910.147 - The control of hazardous energy (lockout/tagout).

#### **6.2.5 PRIORITY**

Critical

## **7 MAINTENANCE & SUPPORT REQUIREMENTS**

Maintenance and support is crucial as this application has to support many users/customers. The project will be documented well and the source code will be available on GitHub with its versions. Customers will be able to provide feed-back which will be helpful to fix bugs.

### **7.1 FIXING BUGS ENCOUNTERED AFTER RELEASE AND CUSTOMER FEEDBACK**

#### **7.1.1 DESCRIPTION**

The application will support customers providing feedback so that the bugs can be tackled. The application will be tested in its earlier stages by the team to prevent bugs.

#### **7.1.2 SOURCE**

Team

#### **7.1.3 CONSTRAINTS**

N/A

#### **7.1.4 STANDARDS**

N/A

#### **7.1.5 PRIORITY**

Low

### **7.2 ALL CODE SAVED ON GITHUB**

#### **7.2.1 DESCRIPTION**

Maintenance and version control will be done through GitHub, a GitHub repository makes it easy for us to keep track of collaborative and personal projects. Each file on GitHub has a history, making it easy to explore the changes that occurred to it at different time points

#### **7.2.2 SOURCE**

Team

#### **7.2.3 CONSTRAINTS**

N/A

#### **7.2.4 STANDARDS**

N/A

#### **7.2.5 PRIORITY**

High

## **8 OTHER REQUIREMENTS**

Quick-quote will be a full-stack application to help provide an easily adjustable format for any future changes in desired functions. This will also help create an application that is compatible with smart-phone web browsing.

### **8.1 REACT JS FOR FRONTEND**

#### **8.1.1 DESCRIPTION**

Our front end website application will be built and maintained using React JavaScript. This is one of the more common open-source tools used for website development today. It offers instant and easy changes to website formatting which will help with the customizable landing page and debugging.

#### **8.1.2 SOURCE**

Quick-Quote Developers

#### **8.1.3 CONSTRAINTS**

Limited knowledge of React and Java Script will require developers to learn and adjust certain parameters based on limits within the React development platform.

#### **8.1.4 STANDARDS**

No current standards limit our front end development

#### **8.1.5 PRIORITY**

Critical

### **8.2 SPRING BOOT FOR BACKEND API**

#### **8.2.1 DESCRIPTION**

Spring Boot is the RESTful service to be implemented as the back end API. It offers an easy to use and easily documented platform for back end development.

#### **8.2.2 SOURCE**

Quick-Quote Developers

#### **8.2.3 CONSTRAINTS**

Back end API must be secure as buyers will use this to enter personal information like credit card numbers and addresses. Limited knowledge of Spring Boot and Java will require much adaption and attention to detail by developers.

#### **8.2.4 STANDARDS**

As a web application HTTP and Transport Layer Security (TLS) protocols will need to be followed.

#### **8.2.5 PRIORITY**

Critical

## **8.3 DATABASE SERVER**

### **8.3.1 DESCRIPTION**

The application will be using MongoDB, an open-source NoSQL database. It will store and organize customer information. The application will be hosted in a live cloud based server after completion.

### **8.3.2 SOURCE**

Quick-Quote Developers

### **8.3.3 CONSTRAINTS**

Team lacks the experience of hosting an application in a live server.

### **8.3.4 STANDARDS**

N/A

### **8.3.5 PRIORITY**

Critical priority

## **8.4 FULL STACK APPLICATION**

### **8.4.1 DESCRIPTION**

It is a full stack project for building complete web application. It involves a frontend, backend, and database element for the application. The frontend is the visible part for user interaction and backend refers to the server-side development.

### **8.4.2 SOURCE**

Quick-Quote Developers

### **8.4.3 CONSTRAINTS**

Team lacks the skills and experience for full stack web application.

### **8.4.4 STANDARDS**

HTTP protocols

### **8.4.5 PRIORITY**

Critical

## **9 FUTURE ITEMS**

### **9.1 QUOTING BASED ON OTHER FILE TYPES**

#### **9.1.1 DESCRIPTION**

The goal of Quick-Quote is to create a product that allows companies to offer a competitive instant quote service. Offering quoting based off several file types will increase the market appeal for this product.

#### **9.1.2 SOURCE**

Quick-Quote Developers

#### **9.1.3 CONSTRAINTS**

Some file types to be considered:

- STEP (.stp, .step)
- SOLIDWORKS (.sldprt)
- STL (.stl)
- Parasolid
- Autodesk Inventor (.ipt)

#### **9.1.4 STANDARDS**

Market trends show these are common files types supported for instant quote cutting services

#### **9.1.5 PRIORITY**

Future