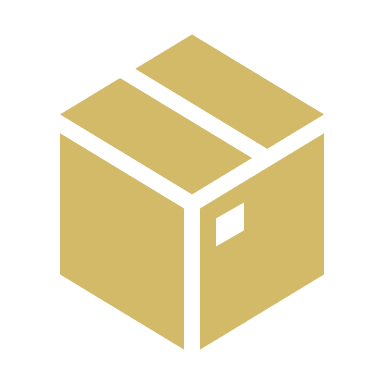
11/25/2019

****

Nathaniel Galbreath nrgalbreath2@mail.fhsu.edu Brandon Nolan btnolan@mail.fhsu.edu Sarah Pauly slpauly@mail.fhsu.edu Logan Wiley lbwiley@mail.fhsu.edu

Order Tracker

Team #5

****Contribution Breakdown****

* Brandon Nolan
* Class Diagram and Interface Specifications
* Design Patterns
* Analysis and Design
* UML Diagram
* Use Case Diagram
* Cover Page/ Individual Breakdown/Design of Documentation
* Table of Contents (Updated)
* References (Updated)
* Glossary of Terms
* Logan Wiley
* System Architecture and System Design
* Architectural Styles
* Mapping Subsystems to Hardware
* Brochure
* Sara Pauly
* System Architecture and System Design
* Persistent Data Storage
* Global Control Flow
* Nathan Galbreath
* System Architecture and System Design
* Identifying Subsystems
* Hardware Requirements
* User Interface Design and Implementation
* Demo Completion

**Table of Contents**

Contribution Breakdown 1

Table of Contents 2

Glossary of Terms 4

Customer Statements 5-6

*System of Requirements* 6

Functional Requirements Specifications 7-12

*Stakeholders* 7

*Actors and Goals* 7

*Use Cases* 8-12

User Interfaces Design and Implementation 13-16

Domain Analysis 17-20

Project Estimation for Use Case Points 21-23

Plan of Work 24

Interaction Diagram 25-28

Class Diagrams and Interface Specifications 29-31

*Design Patterns* 29

*Analysis & Design* 30

*UML Diagrams* 30-31

System Architecture and System Design 32-34

*Architectural Style* 32

*Identifying Subsystems* 32

*Mapping Subsystems to Hardware* 33

*Persistent Data Storage* 33

*Global Control Flow* 34

*Hardware Requirements* 34

History of Work, Current Status, and Future Work 35-36

*Summary of Changes* 36

References 37

**Glossary of Terms**

Administrative – This would be the highest level of permission granted to the computer user within the business. Gives permission to install any further software and make configuration changes to an extent within the program.

Automated Shipment – The process of manual tasks that would be associated with checking and shipping orders in a fulfillment operation put into automatic form for the consumers.

Back Scheduling – The arrival of the package for the customer is calculated at the earliest possible receipt time at the customers unloading point on the requested delivery date. Further adjustment needed to complete customer order on case to case bases.

Customer – This is a person or possible an organization that buys the goods from the business.

Database – An interface that is connected to the website for the customers and is used to store the customers data/ information and tracking information for further use.

Employee – A person employed for wages or salary within this given organization. Carries out jobs in accordance to organizations standards.

Owner – A person or people that own the business.

Package – An object or good that is packed into a box for the consumer upon purchase of this item.

Quote – An estimated price for a service, such as goods, to be given to the customer.

Shipment – The action of the business to ship out goods to the customer. The delivery of the ordered good to the given location from the customer.

Streamlining – A method to make the business more efficient and effective by simpler working methods on employee and consumer side.

System – A set of procedures according to which something is done within the program for the business needs.

**Customer Statement**

It is fair to say that technology and the shipment of goods go hand in hand, as today’s generation drives for ease and convenience. The business has many customers with unique orders that are flowing through the business in different areas. The owners have a current process in place that makes it very difficult for employees to give the whereabouts of a customer’s package. This also leads to many other issues during the delivery process such as when it is going to be delivered, changes in orders, and cancellations. The owners are concerned that by not having proper knowledge on the shipments made, that customers will begin to switch to competitors for consumer needs. While there are numerous automated shipment trackers out there that work; this process can be taken a step forward by implementing more power to the consumers. The system should give more knowledge to the consumer on their products of purchase no matter what time of day. It needs to be a flexible program that can be applied to any consumers electronic because of the variations in smartphones, smart watches, tablets, and laptops. The system should allow the consumer to view quotes of need, orders precise location (including where it has been from start to finish), what teams of employment have worked on it, and see when/why shipments are delayed. This will all be done without the need of interaction from employees for a less complicated/user friendly experience for the consumers.

There are a few assumptions made in order to simplify the system at hand. We assume that the consumer has access to email and an electronic device connected to a network. In addition, it is assumed that no matter the ask of work to be completed, that the quote will always be available to the user.

In order to get the maximum usage out of the system, there will need to be a higher influx of customers. A method to be used is streamlining the new and improved features that give the consumer full access needed to view quotes and shipment locations to perform jobs adequately. In doing so, this will reduce the hassle for customers on their end, increase satisfaction with the business, and therefore increase customer retention. In addition, with the use of this system there will be an attraction for customers because of the ease of use as it pertains to their customer needs. A bonus to the system that could draw more customers in, is the implementation of tracking where a shipment is in accordance to the business teams working on the package.

The website should have an abundance of features that can benefit both customers and the business owner/managers. The website should allow for customers to create personal accounts so that they can store their personal information for future purchases with the idea of having the system be user friendly. The owners/managers will have access to an administrative section of the site in order to set certain variables such as rates on quotes, expediating shipment process, policies, etc. The employees will have their own sperate accounts to handle the day to day operations. Though there is the option of creating a personal account, customers will not be required to create an account but will need a credit card in order to link the shipment to that said customer. The website will add convivence to owners/managers, employees, and customers who will need the service of the business.

Ideally, there will be a mobile application that allows customers to specifically find where their shipment is in accordance with consumer needs. This application will also have the feature of toggling between notifications of the shipments such as: a completion step of one team, shipment location, behind schedule, etc. This will give customer the ease of mind when using the business, creating a smooth user-friendly transaction that will lead to better customer retention.

The system needs to be able to prioritize the quote reports accordingly based on the customers need of date. When an order is received and entered into the system it will be back scheduled based on the customer delivery date or quoted delivery to set an order schedule for the business to follow. The system will allow consumers to view the precise location of the package, along with what team in the business is carrying out that said action. Also, it will track the teams start and completion dates to help maximize efficiency of carrying out orders. The system will also allow users the ability to visually see which orders are behind schedule and the areas that contributed to this delay.

The system will be developed with ease of use for consumers and employees as the main priority. It will need to be intuitive and easy to understand as the business does not have the time to train employees on a unique and comprehensive system. This will also help in drawing new customers in with a simple, yet effective system that gains more retention from the customers. The three main areas of work of the project are: quote/ order entry, quote/ order details, and quote/ order reporting. This idea of quote/ order areas will create simplicity and capture the needed information for that said stage of the business process.

System of Requirements:

* View a prioritized Quote Report based on customer need date
* Back scheduling Orders from delivery date to prioritize workload
* View an orders precise location
* View which business team is carrying out said action
* Track employees and start/completion dates
* Visually see if an order is behind schedule

**Functional Requirements Specification**

***Stakeholders:***

The system will be developed with ease of use as the main priority to have a user-friendly environment. This system will help the business increase profits and have better customer retention. Below are examples of people and organizations who would be interested:

* Business Owner
* Managers
* Employees
* Users

***Actors and Goals:***

|  |  |
| --- | --- |
| Actors | Goals |
| Administrator | To manage quote pricing, override of expediate shipping and analyze statistics. |
| Employee | Handle day to day customer support in unison with system. Update system with completion of work from team to team. |
| User Interface | display open quotes and open orders by priority while not displaying any closed or completed quotes/orders, view an orders precise location and which team has it, track teams and start/completion dates, and see if order is behind schedule. |
| Database | Stores data for the system and help fulfill orders in accordance to prioritizing based on customer need. |

***Use Cases:***

Casual Description –

UC1: Register Account:

* User creates an account to allow for quick access to quote needs. Requires user to create account over a network and provide information.

UC2: Reserve Dates:

* Allows user to reserve dates based on customer date of need for completion.

UC3: Quotes:

* Employees can view quote reports based on scheduling to prioritize workload.

UC4: Location:

* User can view the orders precise location upon shipment from business.

UC5: Team:

* User can view which team is working on that said order. Furthermore, the viewing of the start and completion dates.

UC6: Behind Schedule:

* User can visually see if an order is behind schedule; this being while completing that said quote or during the shipment process.

Fully Dress Description –

\*Use Case 1: Register Account

**Primary Actor:** User, System

**Goal:** To allow users to create an account.

**Stakeholders:** System, Database

**Precondition:** System will request the required information needed from the

customer

**Post Condition:** The customer’s account will then be stored within the

database.

**Main Success Scenario:**

1. Customer goes to the website and chooses the “Register” option.
2. The system then returns the page that states the needed information required.
3. The customer then fills out the required fields of entry.
4. The system takes in the information to verify it.

If not valid, move back to 3. If valid, continue.

1. Information is stored in the database.

\*Use Case 2: Reserve Dates

**Primary Actor:** User, Database

**Goal:** To successfully reserve needed dates for customer.

**Stakeholders:** User Interface, Database

**Precondition:** The user should be logged into their account at this time. The

system will then prompt the user for the reserved dates

needed.

**Post Condition:** The system will put the requested reserved dates into the

database.

**Main Success Scenario:**

1. User enters the dates based on need of completion.
2. System will confirm the reservation date time for completion.
3. The system will then direct the user to a page which displays the reservation has been confirmed for the customer.

\*Use Case 3: Quotes

**Primary Actor:** Employee

**Goal:** To allow employees access to view work orders to prioritize

workload.

**Stakeholders:** Employee, Database

**Precondition:** The system will allow employees to view the work orders that

have been placed.

**Post Condition:** This will give the employees information to allocate

working time according to customer order needs.

**Main Success Scenario:**

1. The employee enters the system to view work orders.
2. System will give work order in order from when date placement is made.

\*Use Case 4: Location

**Primary Actor:** User

**Goal:** To allow the users to view the precise location of their order.

**Stakeholders:** User Interface, Database

**Precondition:** The user will be taken to an interface with their order history

and order made.

**Post Condition:** At the page, the user will be able to view the exact location of

the order they made step by step.

**Main Success Scenario:**

1. User enter their login to sign into their account.
2. Upon login, they will be able to view their current orders they have.
3. The system will give them the option of viewing the exact location of the current product they have ordered.

\*Use Case 5: Team

**Primary Actor:** User

**Goal:** To allow the user to view what business team is working on

order and the completion of each team.

**Stakeholders:** User Interface, Database

**Precondition:** The system will have a selection area for the user to interact

with on the main account page regarding the work

progress.

**Post Condition:** The user will be able to view their work order

made in perspective to the team that is completing it at that

given time.

**Main Success Scenario:**

1. User is prompted to sign into their given account with their specific credentials.
2. The system will load up the user interface and populate an area of work order.
3. In the work order area, user will be able to view all work orders made from earliest order to latest order made with completion dates.
4. Each order will also show what given business team is working on the order at that given time.

\*Use Case 6: Behind Schedule

**Primary Actor:** User

**Goal:** To allow the user the option of viewing if their given order is

behind schedule.

**Stakeholders:** Administrator, User Interface, Database

**Precondition:** The system will populate the users work order for details on

that said order.

**Post Condition:** User will be able to view if the order is behind schedule with

details on that said issue.

**Main Success Scenario:**

1. User is prompted to sign into their given account with their specific credentials.
2. The system will load up the user interface and populate an area of work order.
3. The system will give key details on the order of the user including if it is behind schedule and why this is the case.
4. If behind schedule, then send out alert. If not, then continue forward with work order.

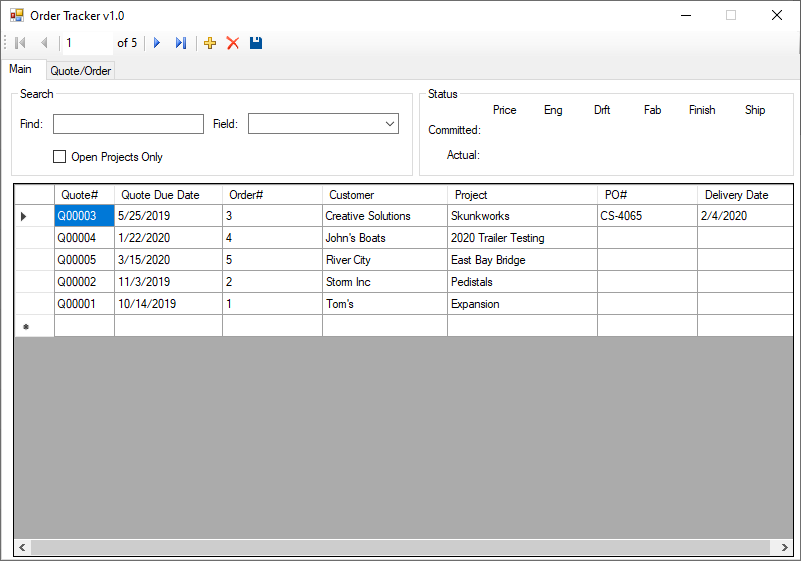
**A close up of a map

Description automatically generated**

**User Interface Design and Implementation**

The user interface is designed is a way to be intuitive and user friendly. Minimal data is needed by the user and one click is all that is needed by the user when “Logging” data for Start/Finish Logs.

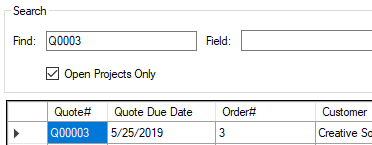
The main display is a grid-based view for simplicity along with a direct Quote/Order visual representation of location when a Quote/Order is selected.



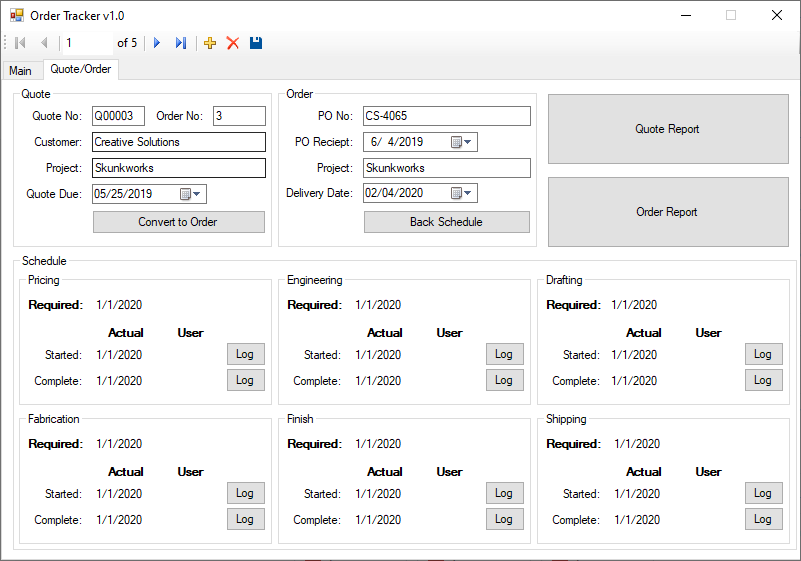
Users will be able to select a Quote/Order and see a Status that shows where the project is located and if it has been tracking on time (Green) or late (Red).



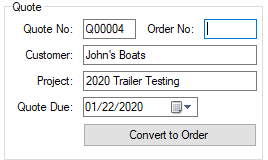
The main screen will also contain a Find or Search allowing the user to quickly narrow down the results to what they are looking for. This will have the ability to limit the Field that is being searched. They can also limit it to only Open Projects.



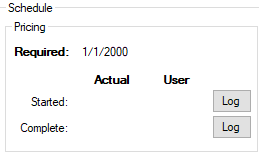
Aside from the main screen there will be a Quote/Order screen used for Quote/Order entry along with a Schedule where users can Log there committed Start and Completion dates into the system.



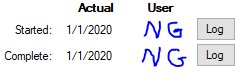
For Quote entry users will be required to enter a Customer Name, Project, and Quote Due date. Quote Number should be auto generated.



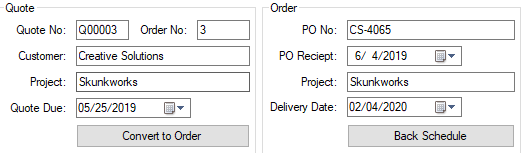
At this point the system will create a quote in the system a set a required date of completion based on the Quote Due date.



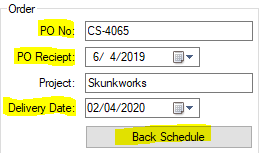
When a User Starts or Completes a quote, they will select the “Log” button to log their date/user for tracking purposes.



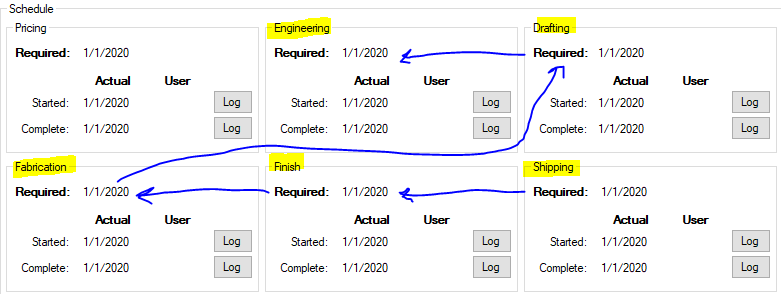
When the user receives a purchase order for the project, they will use the “Convert to Order” button to begin the order entry process. This will assign an Order Number.



The user will be required to enter the customer PO#, PO Receipt date, and Delivery Date for the project. Once logged the user will select the “Back Schedule” button to back schedule the requested dates for the rest of the business users to complete.



Back Scheduling will take the delivery date and schedule Engineering, Drafting, Fabrication, Finish, and shipping to meet the delivery address. Users of each department will Log their time the same way as Pricing for a quote.



The back Schedule and User Log’s is what will drive the visual display for tracking.



The user will also have the option to view a Quote/Order report. This will likely be added as an additional tab and not driven off buttons like in the current display. This will be displayed with a grid view based on open Quotes/Orders prioritized by Quote Due/Delivery Dates. This should display similar to the below with one reporting grid for open Quotes and another reporting grid for Orders

**Domain Analysis**

1. **Domain Model**

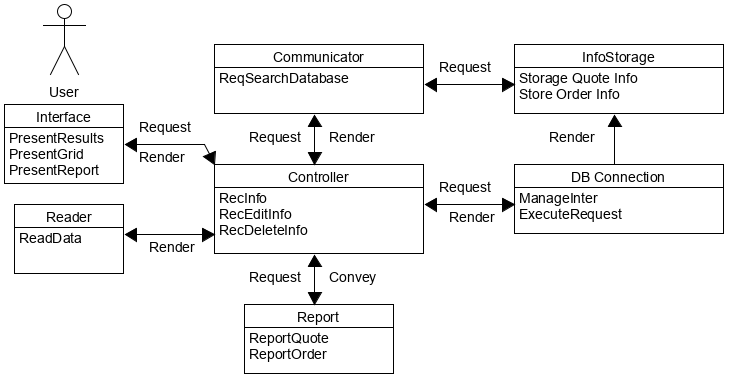


Figure 6-1 Domain Model

Table 6-1 lists the domain model concepts and corresponding responsibilities.



Table 6-1 Conception Definition

**Association Definitions** - Concepts defined above must work in patterns to finish target requirements. Table 6-2 below give Association Descriptions based on defined concepts.



Table 6-2 Association Definition

**Attribute Definition**



Table 6-3 Attribute Definition

**Traceability Matrix**



Table 6-4 Traceability Matrix

**System Operations Contracts**

**Getting Quote/Order Data**

1. PRE-CONDITION – The grid displays
2. POST CONDITION – Users have full view of Quotes/Orders

**Viewing Existing Quote/Order**

1. PRE-CONDITION – The Quote/Order Tab is displayed
2. POSTCONDITION – Users have view of current data for Quote/Order

**Searching Quote/Order**

1. PRE-CONDITION – Main tab is displayed, and Field populated
2. POST CONDITION – User view all Quotes/Orders related to search

**User “Logs”**

1. PRE-CONDITION – Quote/Order tab is displayed, and user has account
2. POST CONDITION – Users “lock” in date/user for current logged task

**Effort Estimation Using Use Case Points**

***Actor Classification:***

|  |  |  |  |
| --- | --- | --- | --- |
| Actor Name | Description | Complexity | Weight |
| Administrator | Manage Quote Price | Complex | 3 |
| Employee | Everyday usage of program | Complex | 3 |
| User Interface | Display quotes, orders, location | Average | 2 |
| Database | Store data for system-fulfill orders | Average | 2 |
|  |  | **Total Actor Weight** | **10** |

***Use Face Classification:***

|  |  |
| --- | --- |
| Use Case Category | Weight |
| UC1: Register Account Complex | 15 |
| UC2: Reserve Dates Average | 10 |
| UC3: Quotes  Simple | 5 |
| UC4: Location Average | 10 |
| UC5: Team Average | 10 |
| UC6: Behind Schedule Average | 10 |
| **Total Use Case Weight** | **60** |

**UUCP = UAW + UUCW = 10 + 60 = 70**

***Technical Complexity Factors:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Technical Factor | Description | Weight | Perceived Complexity | Calculation |
| T1 | Distributed system (Running on multiple machines) | 2 | 3 | 2 \* 3 = 6 |
| T2 | Performance objectives | 1 | 2 | 1 \* 2 = 2 |
| T3 | End- user Efficiency | 1 | 2 | 1 \* 2 = 2 |
| T4 | Simple Internal Processing | 1 | 1 | 1 \* 1 = 1 |
| T5 | Reusable Code | 1 | 0 | 1 \* 0 = 0 |
| T6 | Easy to use- very important | 1 | 5 | 1 \* 5 = 5 |
| T7 | Portable – Not currently necessary | 1 | 0 | 1 \* 0 = 0 |
| T8 | Easy to change | 2 | 2 | 2 \* 2 = 4 |
| T9 | Concurrent use | 1 | 4 | 1 \* 4 = 4 |
| T10 | Security | 1 | 5 | 1 \* 5 = 5 |
| T11 | Direct Access Third Parties | 1 | 0 | 1 \* 0 = 0 |
| T12 | Unique training | 1 | 2 | 1 \* 2 = 2 |
| T13 | Ease of install | 1 | 3 | 1 \* 3 = 3 |
|  | **Technical Factor Total:** |  |  | **34** |

**TCF = C1 \* C2 \* (Technical Factor Total) = (0.6)(0.01)(34) = .204**

**UCP = UUCP \* TCF = 70 \* .0204 = 14.7**

**Plan of Work – Gantt’s Chart**

***Report One:***

* Section 1: Customer Statement of Requirements – Brandon
* Section 2: System Requirements – Brandon
* Section 3: Functional Requirement Specification – Brandon
* Section 4: User Interface Specification – Nathan
* Domain Analysis – Nathan
* Project Size Estimation – Sarah
* Plan of Work – Logan
* References – Logan

***Project:***

* Logan – Front End
* Nathaniel – Front End
* Brandon – Back End
* Sarah – Back End

A screenshot of a cell phone

Description automatically generated

**Interaction Diagram**

***UC1 – User Login:***

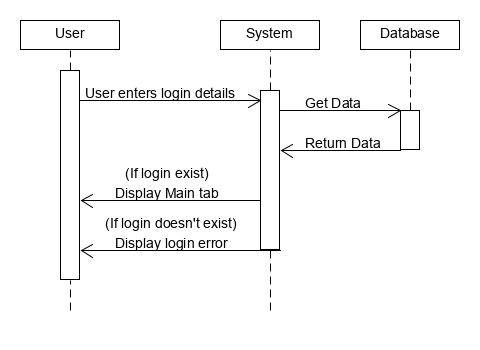


Figure 7-1 Sequence Diagram for Use Case 1

Figure 7-1 shows the sequence diagram for Use Case 1 which is logging in. This is used to monitor who is logging into the system to limit who can and can’t access sensitive information to the company. The user will first enter log in details provided to each employee by the admin (Initials/Password). The system to verify the information against the database and then display either a login error or bring the user to the Main tab of the program and load the data into the grid.

***UC2 – Quote/Order Entry:***

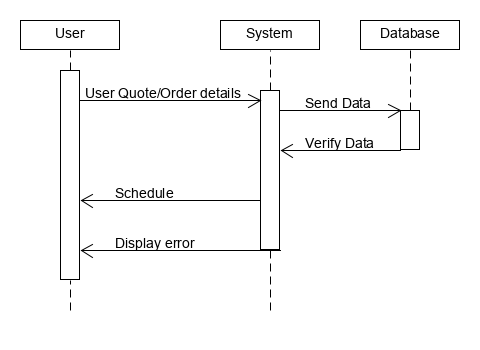


Figure 7-2 Sequence Diagram for Use Case 2

Figure 7-2 shows the sequence diagram for Use Case 2. It is used for getting the Quote/Order information from the user. First the user will enter the Quote/Order information needed and then select “Save”. This will send the data to the database and return a schedule. If there is a scheduling conflict the system will notify the user with an error message.

***UC3 – Reporting:***

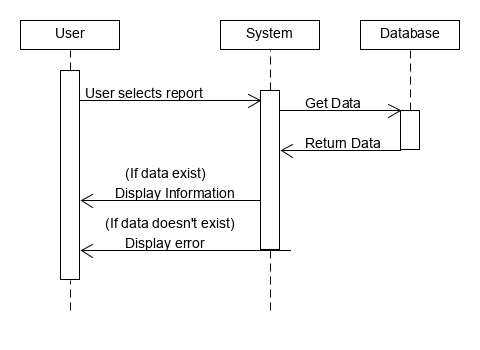


Figure 7-3 Sequence Diagram for Use Case 3

Figure 7-3 shows the sequence diagram for Use Case 3. It is used for getting the Quote/Order reports from the system and display them to the user. First the user will select the Reporting Tab and then select Quote Report or Order Report. This will request and “open” or active Quote/Order data from the database and return organized information to the user. If there are no Quotes/Orders “open” or active, then the system will notify the user with an error message.

***UC4 – Searching:***

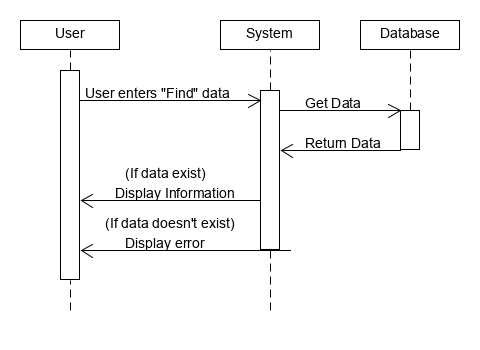


Figure 7-4 Sequence Diagram for Use Case 4

Figure 7-4 shows the sequence diagram for Use Case 4. It is used for searching the Quote/Order data stored in the database and the system will display the results to the user. First the user will enter data into the “Find” text box on the Main tab. A “live” search will begin to narrow the results in the grid for the user to see. If there are no results from the “Find” text box, then the system will notify the user with an error message.

**Class Diagram and Interface Specifications**

***Design Patterns:***

In reviewing the different design patterns, the consensus was that there were two design patterns that most commonly represented our project. The main idea of state design is when an objects behavior depends on its state; it allows the object to change the behavior without changing the class. With this in mind, it should allow for the code to remain cleaner from a visual standpoint for future ease of use. On the other hand, the command pattern was also a pattern that was implemented in our system. This simple means the pattern is intended to encapsulate an object for all the different data require for that said given action or command.

In accordance to our functional features for what we are wanting to accomplish with this system, state design allows the business to view prioritized quote report based on the consumers need of date. This will not only satisfy customer needs leading to more retention for the business, but also will improve workflow for employees and the task to be set forth. This will aid employees to allocate their work in accordance to consumers’ needs with shipment of products. The state design will help our system work hand in hand with employees to make the overall experience of the business more efficient for all parties involved. There will end up being three states here: the order entry, the order details, and the prioritize of time based on customer date needs. The order entry will allow the customer to interact with the system to enter the customer need. The order detail state will be separate but will be the bridge point between the order entry and the allocation of time worked. Order detail will list every piece of information that was listed from the customer in the order entry area. In the prioritizing of time state, employees will be able to allocate time frames based on the information given from the customers. The system will work with the employees to give an understanding of how to allocate workloads based on the dates needed for delivery from the customers order entry and order details. Overall, the state design seems to be the most effective design pattern to implement with our system and aiding the customers and employees with knowing what to do in the different states.

The second design pattern that meshed well with our system was the command design. This design allows the customer to interact with the system for the needed information that they are requesting. This will allow us to implement items/features that will allow the consumers to view updates on their order. It will us to implement and give customers the opportunity to view the orders location, which team is carrying out the current workload at said given time, start and completion times, and if an order is behind schedule. By implementing this into the system, it will hit on our main priority of making it have an ease of use feel for any intended users.

***Analysis and Design:***

|  |
| --- |
| Design |
| Order: |
| -placementDate: Date  -orderNumber: int  -deliveryDate: Date |
| -preciseLocation: Location  -carryingTeam: Team  -startCompletion: Time  -orderBehind: Time |

|  |
| --- |
| Analysis |
| Order: |
| Order Entry  Order Details |
| Prioritize Needs |

***UML Diagrams:***

**State Design:**

|  |
| --- |
| State = PackageState |
| next() |
| prev()  printStatus() |
|  |

|  |
| --- |
| Context = Package Class |
| previousState() |
| nextState() |
| printStatus() |

|  |
| --- |
| DeliveredState() |
| next() |
| prev()  printStatus() |
|  |

|  |
| --- |
| RecievedState() |
| next() |
| prev()  printStatus() |

|  |
| --- |
| OrderedState() |
| next() |
| prev()  printStatus() |
|  |

**Command Design:**

|  |  |  |  |
| --- | --- | --- | --- |
| Customer | Business | Tracking | Customer |
| (client) | (invoker) | (command) | (receiver) |

ViewOrder()

OrderNumber()

OrderPull()

Example:

Order Date

#22361

- Description of

Items

TeamCarry()

OrderLocation()

StartCompletion()

OrderBehind()

**System Architecture**

***Architectural Style:***

Our group used event-driven architecture in the design of our system. An event can be defined as "a significant change in state". For example, when a consumer purchases a car, the car's state changes from "for sale" to "sold". In our case when you change from shipped to delivered by our order tracker.

***Identifying Subsystems:***

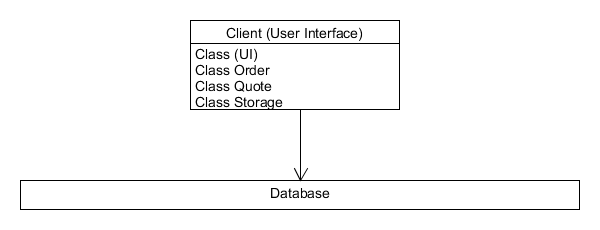


Figure b-1 UML Subsystem Diagram

In this system there are two subsystems. There is the Client subsystem and a Database subsystem. The client subsystem contains the structure for the user interface. In the Database subsystem the Client side sends a request, the request is handled on the Database side, then the result is sent back to the Client side.

***Mapping Subsystems to Hardware:***

The system will have to be to run on multiple devices. In particular, the system will have to be able to communicate with the order tracker. The order tracker will be run on an internal server with multiple computers being able to connect to the order tracker.

***Persistent Data Storage:***

The Order Tracker system must save data so that the data will outlive a single execution of the system and be prepared for the next execution. (i.e. date, time, employee quotes, reports, tracking). The database is used to determine the status of an order that an employee has placed. As we want our system to be error tolerant, we need to make sure order information can be stored and updated in a proper pattern.

The system must store all the information concerning current and past orders or employees in a persistent storage. All operations connecting the information of an order must operate using transactions: at the lower level for updating only one value from the database, relational SQL databases have trans-actionality, for the upper level.

We only want to update the information about an order if the designated order information to be altered has been completely entered by the employee. The simplest method to use is a relational database, MySQL and store the information as tables.

|  |  |
| --- | --- |
|  | Order Tracker |
|  | Pricing |
|  | Engineering |
|  | Drafting |
|  | Fabrication |
|  | Finish |
|  | Shipping |

***Global Control Flow:***

Our system is an event-driven system that waits in loops for events and the user can generate the actions in different order with each use.

The system is an event-response type where there is no real concern for time constraint on the user, besides the scheduling of a time for an order by another employee. The system does not require a time dependency.

When there are multiple users, the integration of their usage will cause concurrency to be an issue. When multiple employees are utilizing the system to schedule orders, the systems database will need to update the storage often to ensure proper scheduling of orders for certain departments to allot.

***Hardware Requirements:***

* Windows based PC
* Minimum 1gb of storage (Expand as needed for database)
* Network connection

The system is built to run on a windows-based PC. For initial installation it is recommended to have at least 1gb of space available for database storage. Storage will need to be expanded as the database reaches its limits. A network connection is required for future updates/syncing.

**History of Work, Current Status, and Future Work**

**During the duration of this project, our team set multiple goals for ourselves in order to work diligent and efficiently so that we would be able to complete our Reports on time in accordance with the due dates. The main idea that revolved around our project was the ease of use so that it was simple in all aspects of aiding the business. The goals we set forth included having a system that aided the customers in tracking/ viewing their packages and aiding the employees in getting the most efficient, effective delivery of those packages.**

**Previous Goals:**

A screenshot of a cell phone

Description automatically generated

**As the project has gone on, we have completed the following:**

* **Viewing a prioritized quote report based on need of date**
* **Back scheduling orders from the delivery date to prioritize workload**
* **Viewing of orders and the precise location as it pertains to delivery**
* **View which team in the business is working on package**
* **Tracking of the start and completion dates**
* **Viewing to see if order is behind schedule**

**With the foundation of what we set to accomplish for the business, others may be able to come along with ease and continue our work by implementing various other expansions. This could be implementing a mobile application for further user-friendly access to the consumers.**

***Summary of Changes:***

* Demo 1 completion and implemented our set goals
* Update of Effort Estimation Using Use Case Points
* Update Table of Contents
* Update References
* Use Case Diagram added
* Glossary of Terms added

**References**

Software Engineering Book

* <https://ww.ece.rutgers.edu/~marsic/books/SE/book-SE_marsic.pdf>

Parking Lot Automation (Rutgers)

* <https://www.ece.rutgers.edu/~marsic/books/SE/projects/ParkingLot/2013-g5-report3.pdf>

Traffic Monitoring Report (Rutgers)

* <http://www.ece.rutgers.edu/~marsic/books/SE/projects/Traffic/2011-g7-report3.pdf>

Gantt’s Chart: “How to Make a Gantt Chart in Excel.” Office Timeline

* <https://www.officetimeline.com/make-gantt-chart/excel#tutorial-auto>.

Conveyco Automated Shipping System

* <https://www.conveyco.com/technology/automated-packaging/dimensional-weighing/>

Microsoft 3D Paint

* Used for some of the design of the diagrams

G-Mail – Used for collaboration with team

* <https://mail.google.com/mail/u/0/#inbox>