Disciples' Pizza Delivery System

System Proposal

For Mr. Park at Disciples' Pizza

Prepared by Cypress Payne at CayCy Development

Executive Summary

CayCy Development has been contacted by Disciples' Pizza to develop an ordering and delivery service for the company's food trucks. The system being developed (Disciples' Pizza Delivery System) needs to support all ends of customer service, from the customer to the chefs and couriers, to the administrators of the company. It will be used to view menu items, place orders, provide delivery instructions, and support administrative changes. DPDS will benefit CayCy Development, Disciples' Pizza, and the community in the Greater Seattle area.

After a thorough investigation into the project's requirements and scope has determined that the project is indeed feasible. Through careful planning, the development of a reasonable timeframe, and addressing the risks associated with the project, the system has high potential for being completed in an affordable and successful manner. CayCy Development and Disciples' Pizza should communicate frequently and create a meeting schedule to ensure all requirements and expectations are met.

1.0 Introduction and Overview

Disciples' Pizza has hired CayCy Development (interchangeably referred to as "CayCy", "us", "we" throughout) looking to develop the Disciples' Pizza Delivery System (DPDS, "the system", "the software", or "the application" throughout) that can support their brand, Disciples' Pizza, as it expands into delivery. DPDS will manage mobile ordering, order display, and delivery instructions.

Section 1 of this document contains a summary of CayCy Development's initial understanding of the system, as well as an assessment of the clients' needs and how we at CayCy will address those areas. This includes the problem statement, project vision and scope, system requirements, stakeholders, expected costs and benefits, project constraints, recommendations, and a document overview.

1.1 Problem Statement

Disciples' Pizza is a group of food trucks that bake wood fired pizza to the greater Seattle area that is hoping to expand into delivery. They contacted CayCy to support their endeavor by building an application, DPDS. DPDS will support the customers, the couriers, and the truck drivers and allow delivery to go smoothly by providing all the needed information in one place. Without this system, delivery of fresh pizza from Disciples' Pizza would be impossible.

1.2 Project Vision and Scope

Disciples' Pizza has a vision of providing fresh pizza and spreading God's blessing around the greater Seattle area. With 30 pizza trucks, and a plan of having 30 to 40 food couriers in the future, they will begin delivering pizza as well. By doing this they can support international missionaries and local churches with the revenue they earn.

Our vision is to create an app for Disciples' Pizza that allows the group to run a successful order and delivery system. This includes online ordering and status tracking on the customer's end, handling

orders and relaying information to the food truck owners, supporting the couriers and allowing them to deliver quickly and accurately, and allowing for menu change and administrative support in order to improve the system's function.

DPDS will require input from external sources to create its output. Customers will enter their order manually through a user interface. The system will charge the customer and send this request to the food truck owner and courier that are nearby. The food truck owner will update the order's progress before passing it on to the courier, who will also input their progress until the order is successfully delivered to the customer. All users will enter their information into the system to create a user profile, although app permissions will vary based on the type of user.

DPDS will support both credit card transactions and the use of food voucher cards from Disciples' Pizza through a promotion code option. It will also support updates and changes to the Disciples' Pizza menus and provide a sales summary and statistics to Disciples' Pizza.

1.3 Requirements Summary

The following is list of major business requirements that DPDS will fulfill:

- DPDS must be able to create new accounts for customers, couriers, and food truck owners, and administrators. Each account type will store its respective information. Only administrators will be able to access sales summaries and statistics.
- The app will have a full menu. This includes allowing the customer to choose dough, size, sauce, cheese, and toppings for their pizza when they order. There will also be options for preset specialty pizzas and the side options of breadsticks, wings, and drinks.
- DPDS will support credit cards as well as Disciples' Pizza food voucher cards for payment and provide online receipts for the customer.
- When an order is placed, DPDS will dispatch the order to the closest truck and courier. The food truck owner will be able to see the customer's name and order. The courier will be able to view their delivery address as well.
- The app will notify the food truck owner of the new order and the customer's name.
- It will also display all ongoing and pending orders to the food truck owner and allow the owner to update the order statuses.
- The customer will need to see the status of their order (among preparation, in oven, and on its way) as it is updated by the food truck chef.
- DPDS will need to display the courier's location to the customer while their food is being delivered. We will need a GPS system integrated into DPDS in order to support this.
- It will also need to support the owner leaving a message of blessing to be received by the customer.
- For the courier, DPDS will need to show a list of delivery tasks and the current destination of their delivery.
- We will need to provide a navigation system for the courier in order to make accurate deliveries.

- DPDS must retain sale information and statistics to provide to the administrator accounts.
- Finally, DPDS must allow administration to make menu changes and handle food voucher information.

1.4 Stakeholders and Interests

CayCy Development has identified the following groups as having an interest in DPDS being designed and completed:

- Mr. Taiwoo Park and the rest of the members of the co-op want a system that can successfully support the delivery of Disciples' Pizza food and can be used by someone at any level of technological skill.
- The owners of the pizza trucks want a more effective way to receive delivery orders and bless their customers.
- Food couriers would like to be able to make more efficient trips and have easily accessible delivery information
- The customers of Disciples' Pizza want to enjoy pizza that is fresh and delivered quickly.
- The international missions and local churches being supported by Disciples' Pizza want a successful delivery system that increases revenue and allows the co-op to provide more support to those in need.
- CayCy Development's developers and systems engineers want a project that is feasible and manageable in the given time schedule.

1.5 Expected Costs and Benefits

1.51 Expected Costs

The creation of DPDS will be costly in the following areas:

- Developmental costs
    - Developers' salaries for the duration of the project
    - Development environment and tools
- Maintenance costs
    - Support from CayCy Development to fix bugs, upgrade features, and provide training on system usage

1.52 Expected Benefits

The creation of DPDS will produce value in the following ways:

- Provide fresh, local pizza that is delivered quickly
- Improve Disciples' Pizza's sale experience
- Increase Disciples' Pizza's revenue
- Increase delivery efficiency

- Make Disciples' Pizza more accessible to the community
- Spread God's love and blessing

1.6 Constraints

CayCy Development expects the project to be limited in its scope by several factors. CayCy Development has taken preliminary steps to mitigate these constraints.

- Devices running DPDS will require internet connection in order to transfer orders and provide navigation.
    - Mitigation: If internet connectivity is lost, DPDS will store the order locally so that it can be sent once the device is reconnected.
    - Recommendation: Make sure company tablets and phones have a solid internet connectivity.
- Disciples' Pizza members have a wide range of computer skills and some may have limited experience on using delivery systems
    - Mitigation: DPDS will contain a help module that provides information on how to use the system
    - Mitigation: CayCy Development will provide basic training on how to use DPDS to Disciples' Pizza members and employees when the system is deployed
- The system must display only the relevant information to each account type.
    - Mitigation: The system will be separated into four modules: Kiosk, Order, Navigation, and Administrative. These will support customers, food truck owners, couriers, and administrators respectively.
- Disciples' Pizza Delivery System needs to be developed cost-effectively.
    - Mitigation: CayCy Development will use existing systems and software when available, allowing DPDS to be developed in less time and using fewer resources.

1.7 Recommendations

CayCy Development has determined that the development of an order and delivery system for Disciples' Pizza will be beneficial for both parties. Mr. Park and his team at Disciples' Pizza should read through the requirements definition (section 4) and ensure that everything they are hoping for the system to be capable of is defined. We would like to recommend that the development of DPDS be initiated immediately, so that we can move forward with the project. A timeline should be set so that the project can continue on schedule and be developed within a reasonable timeframe.

1.8 Document Overview

This document will include six more major sections:

2 System Initiation

- The received system proposal and CayCy's response letter

3 Feasibility Assessment
- A detailed analysis of the feasibility of the project as whole, as well as the feasibility of specific areas in the system itself.

4 Requirements Definition
- An overview of DPDS' system and behavior

5 Requirements Model
- Use case diagrams and descriptions based on the requirements

6 System Evolution
- How CayCy will support the system's future requirements and upgrades

7 Conclusions and Recommendations
- Summary of the system proposal and important requirements of DPDS as well as a recommendation for how Disciples' Pizza should proceed

2.0 System Initiation

2.1 System Request

October 11, 2019

SYSTEM REQUEST – Disciples' Pizza

Project Sponsor

Name: Mr. Taiwoo Park
Representing: Disciples' Pizza
Phone: x7258 E-mail: twp@spu.edu

Opportunity Statement:

Disciples' Pizza is the new brand of the co-op of Christian wood fired pizza trucks in the greater Seattle area. Our trucks have our own unique style and taste and thus been loved by local residents for years. Our trucks move to many different places every day, so it has been challenging for us to provide our customers with a quality delivery service. However, we would like to start a delivery service by hiring food couriers (i.e., delivery persons). It is a very interesting setting – our kitchens and food couriers both move! We are going to need some technology to make *this* happen!

Proposed Product:

Background and Context:

Disciples' Pizza isn't a "company" per se, but rather a group of food trucks and a group of food couriers to bake and deliver pizza to local customers. We have 30 pizza trucks in the greater Seattle area, mostly on the west side of Lake Washington, and will have 30-40 food couriers in the near future. Most of our trucks moves every day, or even in the middle of a day depending on neighborhood and/or sports events. For its brand and to support our upcoming mission of international missionary and local church support, we would like to have our own delivery system and mobile apps for online orders.

We would like to take a full advantage of mobile devices for trucks, food couriers, and customers. Specifically, most of trucks already have tablets for credit card transactions. Food couriers have smart phones, and of course, our customers do. Our dream is that once our customer is craving for pizza, s/he opens our app, and make an order, then the nearest pizza truck receives the order. Also, one of our food couriers nearby will receive a delivery request, and the courier will deliver the pizza to the customer. We hope that all these processes would work smoothly to find the best truck and courier so that we can be always cost-effective and maximizing customer satisfaction.

Initial Vision and Scope:

- *Online Order and Status Tracking*

  Our customers need to be able to order pizza specifying its dough, size, base sauce, cheese, and toppings. We also offer a set of specialty pizzas with presets, as well as bread sticks, wings, and drinks. We think to support credit cards as main payment method, while supporting our own food voucher cards for ones who are in need. Once a customer makes an order from our app, it automatically dispatches the work to one of our trucks --- perhaps the closest one -- and one of our couriers. Our app needs to be able to show the current status of the order, among preparation / in oven / on its way, and the courier's location in case when the pizza is on its way. After delivery, our customers can hear a message of blessing from the truck owner who made the pizza.

- *Pizza Truck Order Handling*

  The app in our food trucks is to notify our food truck owners of new orders and show all ongoing and pending orders on the screen. Most of our food truck owners make multiple pizzas at the same time, while handling customers in-person, so the app needs to give the information as clear as possible. Our owners are supposed to change the current status of the orders, and it would be nice if they can record a voice message (to spread the love and blessing of God) for a customer when they hand the pizza to a courier. Also, our owners should be able to create a new order made in-person through the app.

- *Courier Support*

  Our couriers will have a smartphone app for delivery information. It will show a list of delivery tasks, as well as the current destination. If a courier is with pizza, the destination is to be delivery address, or a pizza truck for the next delivery otherwise. Our couriers need navigation service for their pickup and delivery.

- *Menu change and administration support*

We may want to change ingredients and specialty pizza styles depending on availability and season, hopefully through web browsers. Also, we would like to handle the food voucher information.

- *Sales summary and statistics*

Stakeholders Identified:

- Pizza truck owners – who would like a more effective way to receive delivery orders and bless the customers.
- Food couriers – who would like to make more efficient trips and easy 'where-to-go-next' information.
- Our customers – people who will enjoy our fresh pizza delivered quickly.

- You – as our partner.

Expected Benefits:

- Opportunity 1 – Fresh pizza, made in my neighborhood, delivered quickly.
- Opportunity 2 – Improve our sales experience and increased revenue.
- Opportunity 3 – Spread God's love and blessing.

Special Issues or Constraints:

We're not made of money. Members are willing to invest in the new equipment and programs, but we're talking a few hundred (not thousand) dollars each from maybe 30 owners. Similarly, I'm not sure everyone is going to be excited to run out and by a new phone or tablet to manage sales or use this system. (Although maybe we could handle that with a few purchases "by the co-op".)

Also, our members represent a wide range of computer skills and types of computers they know and use. Some are really into their smartphones and others aren't. Pretty typical people.

I don't think we are on a particularly tight time schedule. The core sales functionalities might be considered to be finished sooner, while others can be more down the road.

2.2 Sales Letter

Caycy Software Development, Inc.

10/21/2019

Mr. Taiwoo Park, Project Sponsor
Disciples' Pizza
241 Miller St.
Seattle, WA 98119

Dear Mr. Park,

I was thrilled to receive your system request regarding an app needed for Disciples' Pizza. Your idea of having food trucks have a mobile delivery service is innovative and has lots of potential, especially in a busy, tech-focused city like Seattle. This, as well as the focus on blessing customers and personalizing their experience makes me sure that this co-op has promise and will go far.

As a company with previous experience in food and delivery apps, I am positive that we at CayCy can make your dream happen. In your request, you describe your need for an app that can support the customer, the truck owners, and the couriers. We at CayCy have worked with restaurants in the past and ensured that their mobile order and delivery app runs smoothly for both customers and the company themselves.

I also understand the concerns that you and your members have brought up, and I hope to allay these. I have spoken with several members of our team here at CayCy and they believe that your specifications can be matched without spending too large a sum of money. The applications that CayCy produces are made to be used on most smart devices. We also test our products to ensure that they are easy to use and meet customers' expectations. CayCy has worked with many different companies to create the apps that they are looking for, and we pride ourselves in taking all our partners' visionary ideas into account.

I will be in touch within the next week to discuss our plans moving forward. Do not hesitate to reach out sooner if you have any questions or concerns. I look forward to speaking with you.

Sincerely,
Cypress Payne
CayCy Software Development, Inc.

3.0 Feasibility Assessment

3.1 Introduction

This section provides a summary of CayCy's feasibility analysis for DPDS. Our feasibility assessment determines the practicality of the project as a whole given the time and budget constraints. This is done by examining the feasibility of five subcategories: Technical, Resource, Schedule, Organizational, and Legal/Contractual. Each subcategory is assigned a feasibility rating and a risk rating.

The risk ratings are as follows:

- **Low:** There is minimal risk associated and the probability of issues occurring is very low.
- **Moderate:** There is moderate risk associated and some adjustments may need to be made.
- **High:** There is a concern that must be addressed, or the consequences are serious.
- **Very High:** There is high risk that must be addressed before the development of DPDS can continue.

Additionally, the feasibility ratings are as follows:

- **Infeasible:** The risks in this category are too great to justify the project continuing.
- **Feasible:** There are potential challenges, but they can be overcome with planning and consideration.
- **Ideal:** There is very little risk in this category, and it will greatly benefit the project.

3.2 Feasibility Analysis

3.2.1 Technical Feasibility

FEASIBILITY: FEASIBLE	RISK: LOW

- User Familiarity
    - There is some risk associated with user familiarity. Mr. Park of Disciples' Pizza has informed us that members' technological skills vary. This needs to be considered in the development of DPDS. CayCy will take measures to train the users of the system on the basics when it is deployed.
- Analyst and Developer Familiarity
    - CayCy has past experience in projects similar to the Disciples' Pizza Delivery System. Our developers and designers have built functional and user-friendly systems designed for customer service before and we are confident in our understanding of the requirements this system has from a technical view.
- Project Size

    - There is little concern with the project size at this point. The system is a basic computer and mobile application. It will not require much storage or bandwidth, although internet access will be needed.
- Project Structure
    - We have a clear idea of the requirements that DPDS will need to meet, provided to us by the client. There is not much risk that these requirements will change, although additional requests may be considered as the project moves forward.

3.2.2 Resource Feasibility

FEASIBILITY: IDEAL	RISK: LOW

- CayCy has assigned a team of 12 experienced personnel to move forward with the project. All of these employees have had previous involvement in similar projects
- There is very little hardware required for this project. DPDS is being designed to be used on any smart device.
- CayCy Development has software available that will be used to develop the project.
- There is some risk associated with the project's environment. DPDS will retain some personal information from the users but it will be protected from the applications' environment.

3.2.3 Schedule Feasibility

FEASIBILITY: FEASIBLE	RISK: MODERATE

- The client did not give a strict timeframe and we currently do not have any completion dates; however, they have stated that core sales functionalities is the top priority timewise. It is difficult to meet milestones with no set dates, so there is moderate risk associated here.
- To prevent slippage, setting a strict timeline will be necessary and ensure that the project is finished by any deadlines that will be set later.

3.2.4 Organizational Feasibility

FEASIBILITY: IDEAL	RISK: LOW

- The project champion, Mr. Park, as well as the rest of the members of Disciples' Pizza desire to see this project succeed. It will aid in Disciples' Pizza's goal of quick delivery of fresh pizza to the community.
- The food truck owners and couriers will be the primary users on the service end of DPDS. They will see an increase in efficiency and ease of delivery when the system is implemented.
- Disciples' Pizza customers will be quick to adopt the system, as it will allow them to order fresh pizza to their homes and businesses from their favorite food trucks.
- CayCy is invested in the system because it is the best solution for Disciples' Pizza and will allow them to see an increase in efficiency, customer satisfaction, and revenue.

3.2.5 Legal and Contractual Feasibility

FEASIBILITY: FEASIBLE	RISK: MODERATE

- DPDS will require a navigation and GPS system. This resource will not be built by CayCy but will be acquired elsewhere, meaning that there is some risk as it will not be the property of CayCy or Disciples' Pizza. We will need to pursue the correct channel to acquire this navigation resource in order to minimize legal risks.
- CayCy will retain rights to DPDS. A contract will need to be drawn up between CayCy Development and Disciples' Pizza. This is low risk but will need to be dealt with.

3.3 Additional Comments

- This project is a good fit for CayCy considering our previous experience.
- A tutorial video or training session may be a good idea in order to ensure that all service-end users understand how to use DPDS

3.4 Conclusion

While there is some risk with developing DPDS, CayCy has determined that the project is **feasible** with a **moderate-low** risk factor associated with it. With risk management and mitigation, the project will be successful. CayCy Development recommends going ahead with project development.

4.0 Requirements Definition

4.1 Introduction

This section defines the features and characteristics of DPDS. This includes functional requirements – features that DPDS need in order to run properly, and non-functional requirements – the limitations and behaviors of the system. The system's performance is divided into five sections: User Accounts, Kiosk, Order, Navigation, and Administration. These are based on the display that will be visible depending on whether the user is a customer, courier, food truck chef, or administrator using the system.

4.2 Functional Requirements

4.2.1 User Accounts

- DPDS must allow for four different types of account creation. Anyone may create a customer account, but courier, chef, and new administrator accounts must be set up by a current administrator.
- The system must require that customer accounts input their name, phone number, email account, and address as well as create a username and password in order to access their accounts.
- Courier, chef, and administrator accounts should require their name and email to create a username and password.
- The system must allow administrators to create, modify, or delete any accounts on the system.
- Access to modules must depend on account type. Each account type will have access to these modules:
    - Customer: Kiosk
    - Chef: Order
    - Courier: Navigation
    - Administrator: Kiosk, Order, Navigation, and Administrative

4.2.2 Kiosk Module

- The kiosk module must display the menu so that the customer can place an order. The menu will be displayed as buttons for each item that the customer can then click to add to their cart.
- The system must allow the customer to view their cart with the menu items they have chosen and edit or delete these items.
- At the cart screen there must also be the option to proceed to check out which will charge the customer using the payment method and send the order to the chef and courier to be prepared and delivered.
- This module must require a payment method at check out.

- The system should update the customer on their order's status and location through this module.
- The system should notify the customer if they have received a recorded message from the chef when their delivery is complete and allow them to view it.

4.2.3 Order Module

- The system must send the order from the kiosk module to the order module of a chef based on distance between the location of the truck and the delivery address.
- Information that must be sent includes the food ordered, the customer's name, and the time the delivery was placed.
- The order module should display the customer's name and order based on time the order was placed to the truck's chef and prompt the chef to confirm the order.
- The system must have the option for the chef to update the status of the order as 'being prepared', 'in the oven', and 'on its way' through this module and leave a recorded message for the customer.
- The system should allow the chef to view the name of the courier who will be making the delivery.

4.2.4 Navigation Module

- The system must determine the closest courier and send them the customer's name, order, and address through this module.
- The navigation module must direct the courier to the customer's location of delivery.
- The system must provide clear instructions in order to allow the courier to make a fast, accurate delivery.
- This module must allow the courier to confirm the delivery of the order when it has arrived at its destination so that the system may close the order.
- The system should allow the courier to call the customer if there are any complications.

4.2.5 Administration Module

- The administrative module must allow the user to create, edit, and delete and menu items.
- This module will provide access to unprotected user data (name, account activity) so that the administrator can create, edit or delete user accounts.
- This module also opens the User Accounts module in administrative mode, allowing for courier, truck chef, and other administrator accounts to be created.
- Sales summaries and statistics will be generated and made visible in this module.

4.3 Data Requirements

This section describes the data that must be stored within the system, as well as how this data can be accessed and constraints that that data has.

4.3.1 User Accounts

Information Stored:

- User Account Type: Customer, Chef, Courier, or Administrator
- First and Last Name
- Email Address: The user's unique email address
- Username: a unique identifying name with no spaces or special symbols
- Password: a string of characters that grants access to the account
- Address: The user's street name, city, state, country, and zip code
- Phone number: The user's phone number consisting of 10 numerical characters
- Credit card information: The user's credit card number, expiration date, and security code

Constraints:

- Every user account must have an account type, first name, last name, and email address. These will be entered by the user when the account is first created.
- Client accounts must have an address and phone number.
- Chef, Courier, and Administrator account usernames will be generated by the system based on the user's first and last names. These can only be changed by an administrator.
- User's passwords can be changed at any time.
- User Account information will be stored until an administrator deletes the account.
- Only administrators can delete User Accounts.

4.3.2 Menu Items

Information Stored:

- Item category: Dough, Size, Sauce, Cheese, Topping, Specialty, or Size
- Item name
- Description
- Price: Numeric value in USD

Constraints:

- Only administrators can add, edit, and delete menu items.
- Each item must have a name, category, and price.

4.3.3 Transactions

Information Stored:

- Date: the hour, day, month, and year that the order was placed.
- Time ordered: the minutes and seconds the order was placed.
- Time received: the minutes and seconds the order was received.
- Items: The menu items that were ordered

- Cost: The total cost of the items
- Customer: The customer account that made the purchase.
- Chef: The chef account that was assigned to cook the order.
- Courier: The courier account that was assigned to deliver the order.

Constraints:

- Transaction data will be recorded automatically by the system as orders are completed.
- Transaction data will be stored in the system for the duration of the system's life.
- Transaction data will be available to be viewed by Administration Accounts.

4.4 Nonfunctional Requirements

4.4.1 Operational Requirements

- DPDS must be able to run on multiple, separate computing devices simultaneously.
- DPDS must be able to have multiple accounts running and orders occurring simultaneously.
- DPDS modules must be clearly divided and only accessibly by their related account types.

4.4.2 Performance Requirements

No specific performance requirements are anticipated.

4.4.3 Security Requirements

- Only Administrator accounts will have access to user account information.
- Credit card information will only be accessible through the account it is stored in.
- All accounts must be password protected.
- Daily backups of the account, menu, and transaction databases will be performed.

4.4.4 Cultural Requirements

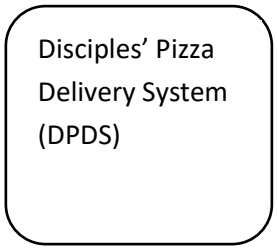
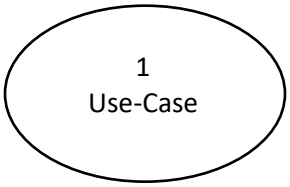
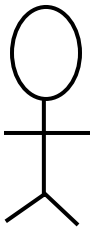
- DPDS will originally only be available in English.
- More languages could be made available in future versions.
- DPDS will only be serving the Greater Seattle Area.

5.0 Requirements Module

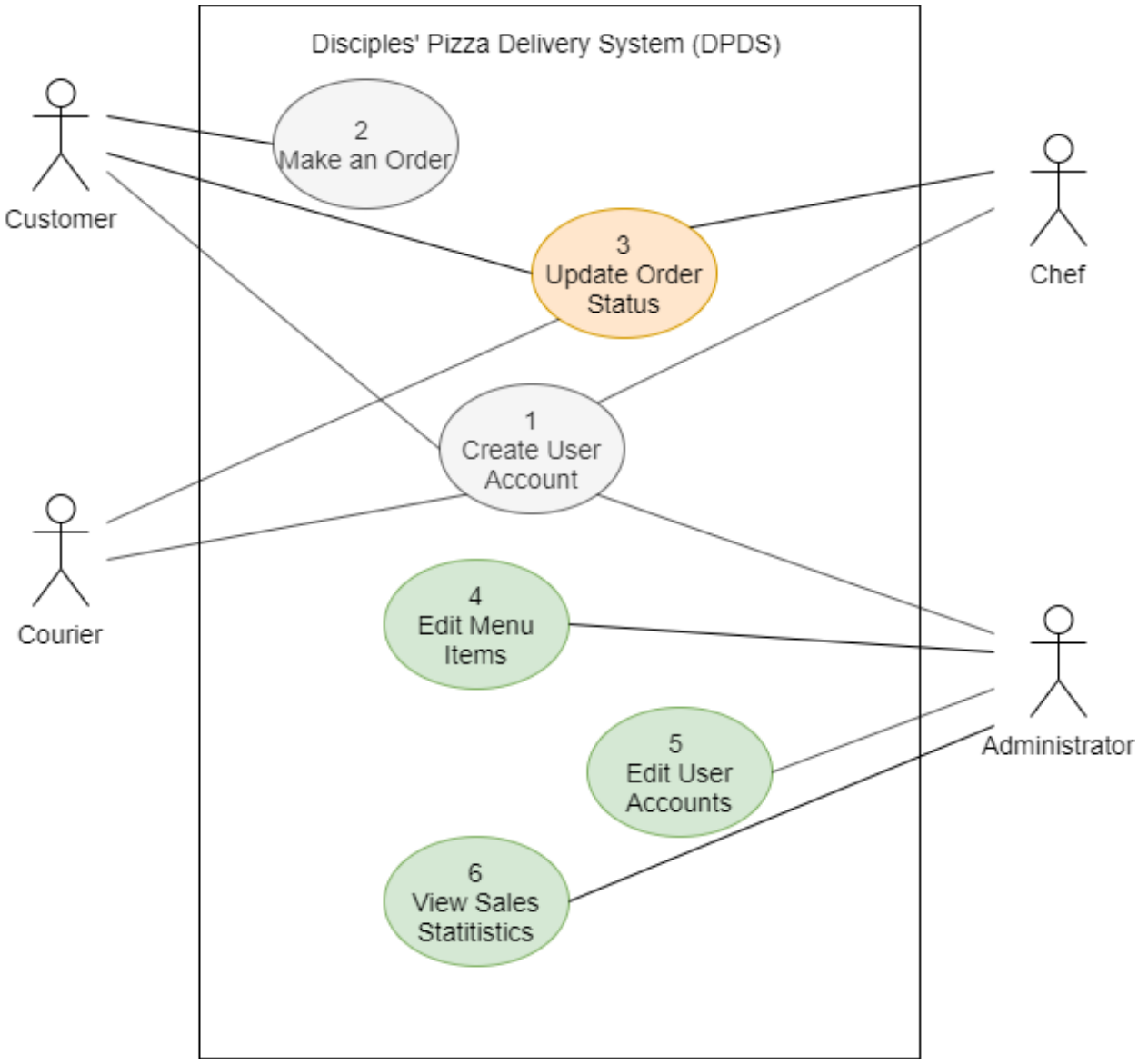
5.1 Introduction

The following sections contain the Use-Case Diagram (5.2) and Use-Case Descriptions (5.3) of the system. Each use-case is a foreseeable defined interaction between the users and the system. The Use-Case Diagram maps out each use-case and the users involved. For more detail into each use-case please see the Use-Case Descriptions (5.3).

The Use-Case Diagram's elements are defined in the following legend.

Name	**Graphic**	**Description**
DPDS		The square represents the system, and all use-cases are contained inside of it. Actors are outside the system and are drawn outside of the square.
Use-Case		A Use-Case is a series of actions that is likely to be taken by an actor to accomplish a task in the system. Each use-case is labeled with a number and name and is described in detail in 5.3.
Actor		An Actor is a person who has direct influence on the system and provides input that triggers the use-case to start.
Association		An Association connects an actor to a use-case that the actor's input will execute.

5.2 Use-Case Diagram



5.3 Use-Case Descriptions

Use-Case name: Create User Account	**ID**: 1	**Importance**: High
Primary actor: Customer, Administrator, Chef, Courier	**Use-Case type**: Detailed, Essential	
Stakeholders and interests: A Customer wants their information to be saved by the system to make ordering quick and easy. An Administrator wants their accounts to be private and secure. The Chef and Courier want to create a user account so that they can view order and destination information.		
Brief description: Creating a user account occurs when the user first accesses the system. This information is then saved so that the user can access their account in the future. The user must have an account to access the module associated with their user-type.		
Trigger: The actor selects to create a new account when they access the system. **Type**: External Temporal		
Relationships: **Association**: Customer, Administrator, Chef, Courier **Include**: None **Extend**: None **Generalization**: None		
Normal flow of events: 1. Actor selects Create New Account on a DPDS Application or website. 2. Actor selects account type they want to create: Customer, Chef, Courier, or Administrator. 3. The system prompts the user to enter the necessary information to create an account. 4. DPDS saves the account information so that the account can be accessed in the future.		
Subflows: S3 – Enter Personal Information 1. All account types require the user's name, email, a username, and a password. 2. If the account type is customer, the user must also enter their phone number and delivery address.		
Alternate / exceptional flows: 1a. If Actor selects to Login instead of Create New Account, they will be prompted to enter their username and password so that they can access their already existing account. 2a. If Chef, Courier, or Administrator account type is chosen, the account must receive administrator approval before the account is created. 3a. If the data entered is not valid or complete, the system will display an error message. Return to step 3.		

Use-Case name: Make an Order	**ID**: 2	**Importance**: High
Primary actor: Customer	**Use-Case type**: Detailed, Essential	

Stakeholders and interests:
A Customer wants to order their food in a straight-forward manner.
A Chef wants this order to be made so that they can make the food and keep customers happy.
The Disciples' Pizza Co-op Members want to keep customers happy and receive mobile orders to increase revenue and customer access.

Brief description: The customer will select what they want from the menu and pay for their order. This order will then be created in the system and sent to a chef and courier in order to make the order and deliver it to the customer.

Trigger: The customer selects the items they want from the menu on the DPDS app.

Type: [External] Temporal

Relationships:
- **Association**: Customer
- **Include**: None
- **Extend**: None
- **Generalization**: None

Normal flow of events:
1. Customer selects items off of the menu, which DPDS stores as the customer's cart.
2. The customer then proceeds to check out when they have selected everything they want.
3. DPDS calculates the total of all items in the customer's cart.
4. DPDS confirms the customers delivery address and payment method.
5. The system charges the customer using the payment method.
6. The system sends the order and delivery information to the nearest chef and courier so that the order can be completed.

Subflows: None.

Alternate / exceptional flows:
4a. If the customer does not have a payment method, or there is an error with their payment method, DPDS will display an error message and prompt the customer to update their payment information. Return to step 4.

Use-Case name: Update Order Status	**ID**: 3	**Importance**: High
Primary actor: Chef, Courier	**Use-Case type**: Detailed, Essential	

Stakeholders and interests:
Both the Chef and Courier want to update the customer on their order and keep the customer happy.
A Customer wants to be informed about their order and reassured that it is being made to specification.

Brief description: The Chef and Courier will select the status of the order they are working on. When this is updated, the order status will be sent back to the customer through the system.

Trigger: An order is sent to the chef and courier.

Type: External Temporal

Relationships:
Association: Chef, Courier
Include: None
Extend: None
Generalization: None

Normal flow of events:
1. DPDS notifies the chef and courier of a new customer order.
2. The chef and courier view the order and update the status.
3. The status is sent back to the customer.

Subflows:
S2 – Update the status
1. If the account that receives the order is a chef account, they can select from three statuses: ‘Being Prepared’, ‘In the Oven’, and ‘On its Way’. They can also record a message that will be sent to the customer.
2. Once ‘On its Way’ has been selected, the Courier is allowed to update the status to ‘Delivered’ when the delivery has been made.

Alternate / exceptional flows: None.

Use-Case name: Edit Menu	**ID**: 4	**Importance**: High
Primary actor: Administrator	**Use-Case type**: Detailed, Essential	

Stakeholders and interests:
An Administrator and a Chef want accurate menus that the Customer can select from and the chef can cook.
A Customer wants an accurate and updated menu so that they will receive what they have ordered.

Brief description: The Administrator can edit the menu that is viewable to the customer through the system. They will be able to create, delete, and edit previously existing items so that the menu is as accurate as possible.

Trigger: The Administrator selects the option to edit the menu.

Type: [External] Temporal

Relationships:
- **Association**: Administrator.
- **Include**: None
- **Extend**: None
- **Generalization**: None

Normal flow of events:
1. An administrator selects to edit the menu.
2. They select from the menu editing options: ‘create’, ‘edit’, or ‘delete’.
3. DPDS prompts the Administrator to save and update the menu based on these changes.

Subflows:
S2 – Select from editing options
1. If ‘create’ is chosen the administrator will input the name of the item, the type of item (dough, sauce, cheese, topping, specialty or side), a description, and the price.
2. If ‘edit’ is chosen, the administrator selects an item from the menu and can edit the name, type, description, or price of the item.
3. If ‘delete’ is chosen, the administrator selects an item from the menu and the DPDS system deletes it from the menu database.

Alternate / exceptional flows:
3a. If the Administrator chooses to not save their changes the menu remains the same.

Use-Case name: Edit User Accounts	**ID**: 5	**Importance**: High
Primary actor: Administrator	**Use-Case type**: Detailed, Essential	

Stakeholders and interests:
An Administrator wants to be able to delete accounts that are being misused as well as edit company accounts when necessary and view account activity.
A Customer wants their account to run well and be able to receive help if they have issues with their account.

Brief description: An administrator is able to select an account and edit information if there is an issue. They are also able to delete accounts.

Trigger: An Administrator selects to edit an account in the system.

Type: External Temporal

Relationships:

Association: Administrator
Include: None
Extend: None
Generalization: None

Normal flow of events:

1. An Administrator selects an existing account in the system from the system's account database.
2. An Administrator can select to edit or delete the account.
3. DPDS prompts the Administrator to save and update the accounts database based on these changes.

Subflows:
S2 – Select from editing options

1. If 'edit' is chosen, the administrator is able to view and edit the account's name, username, email, and password.
2. If 'delete' is chosen, the account and information will be deleted from the database.

Alternate / exceptional flows:

2a. If the Administrator chooses to not save their changes the account remains the same.

Use-Case name: View Sales Statistics	**ID**: 6	**Importance**: High
Primary actor: Administrator	**Use-Case type**: Detailed, Essential	

Stakeholders and interests:
Both the Administrator and the Disciples' Pizza Co-op Members want to be able to view sales statistics and see how well the company's delivery system is functioning.

Brief description: The Administrator will be able to view a sales summary and statistics on what customers have been ordering and the profit that Disciples' Pizza has made.

Trigger: The Administrator selects to view sales statistics from the Administration Module.

Type: [External] Temporal

Relationships:

Association: Administrator
Include: None
Extend: None
Generalization: None

Normal flow of events:

1. An Administrator select to view the sales statistics in the Administration Module.
2. DPDS loads the sales statistics that have been generated for the Administrator to view.

Subflows: None.

Alternate / exceptional flows: None.

6.0 System Evolution

Version 1.0 of DPDS will include everything that the system needs to function for Disciples' Pizza. However, new features can be added and released in later versions of the system, including options for other languages or other concerns that are brought up by the co-op. Version 1.0 will contain all requirements containing the word 'must' in section 4.2. Requirements listed as 'should' or 'could' may be postponed until later versions depending on time constraints.

CayCy Development will provide continued support for maintenance and changes to the system after it is deployed. Disciples' Pizza can contact CayCy with ideas for new changes to the app at any time. These changes will be evaluated, and it will be determined if they can be completed in a reasonable amount of time. CayCy will also provide maintenance support to keep the system running smoothly and functioning as expected while the system is in use.

7.0 Conclusions and Recommendations

7.1 Conclusions

CayCy Development will support Disciples' Pizza's goal of spreading pizza and God's love throughout the Greater Seattle Area. DPDS will enable Disciples' Pizza to deliver pizza smoothly and efficiently. CayCy has determined that this project is feasible and can be completed within the timeframe and budget with the available resources. DPDS will be able to support delivery orders, menu changes, customer information, and sales statistics. We at CayCy believe that DPDS is what Disciples' Pizza needs in order to have a successful delivery system. We urge you to move forward with this project.

7.2 Recommendations

With these conclusions in mind, CayCy Development recommends that Disciples' Pizza proceeds with the following steps to ensure that the development of DPDS will run as smoothly as possible.

- Ensure that all food trucks have a working tablet or computer capable of running the system on it.
- Staff an adequate number of couriers for the delivery system to succeed.
- Communicate frequently with CayCy Development about any questions or concerns about the system. Bring up any concerns about the system with CayCy, it is important to us that the system meets all expectations and that the customer is satisfied.
- Meet frequently with CayCy Development's team to view progress and test usability.
- Develop a time schedule so that accountability can be maintained and CayCy can set milestones to be reached.

8.0 Appendices

8.1 Meeting Notes

Meeting with Mr. Taiwoo Park, Representative of Disciples' Pizza.

- Couriers and Drivers will be selected based on distance from delivery address
- Food truck drivers move around.
    - Possibility for the option of viewing driver locations in the future.
    - Possibility of scheduling driver location using DPDS in the future.
- Disciples' Pizza does not have a preference on whether the system is contained to one application or has multiple applications depending on the user.
- No current time constraints are known, although Disciples' Pizza would like the system to be completed in a timely manner.

9.0 Glossary

Administrator: A user of DPDS who is a part of Disciples' Pizza management. They will have access to DPDS' data and user accounts.

CayCy Development: The company who is building DPDS for Disciples' Pizza.

Chef: A Disciples' Pizza co-op member who owns a food-truck and cooks pizza.

Courier: A Disciples' Pizza co-op member who will be delivering pizza.

Customer: A user of DPDS who is not a part of Disciples' Pizza but is a consumer of the company.

Disciples' Pizza: A co-op of food truck drivers serving the Greater Seattle Area who are interested in developing a system for delivery.

DPDS: The Disciples' Pizza Development System. This is the system that CayCy will develop to support Disciples' Pizza's delivery plans.

Stakeholder: Anyone who will be affected by the system and its success or failure.

10.0 Bibliography

Dennis, Alan, Barbara Haley Wixom, David Paul Tegarden, and Elaine Seeman. System Analysis and Design: An Object-Oriented Approach with UML. Hoboken, NJ: Wiley, 2016. Print.

Park, Taiwoo. *Systems Design*, various lectures. CSC 3150. Seattle Pacific University, Seattle. Lecture.

Pfeiffer, William S. Pocket Guide to Technical Communication. Upper Saddle River, NJ: Prentice Hall, 2011. Print.

Various Authors, “5k Foods Sys Prop - 1”, “5k Foods Sys Prop - 2”, and “5k Foods Sys Prop – 3”. CSC 3150. Seattle Pacific University, Seattle. PDF.