# 

**Pizza Connection**

Development Plan

Team Members: Matthew Krol, William Esparza, Mehedi Zihad, Kanta Islam

Table of Contents

[1.1 Project Overview 3](#_Toc145589881)

[1.2 Project Purpose, Scope, Objective 3](#_Toc145589882)

[1.3 Team Organization (Roles and Responsibilities) 5](#_Toc145589883)

[1.4 Problem Resolution Policies 6](#_Toc145589884)

[1.5 Project Plan (iterations, project schedule) 7](#_Toc145589885)

[1.6 Configuration Management Plan 8](#_Toc145589886)

[1.7 Technologies 8](#_Toc145589887)

## **1.1 Project Overview**

Pizza Connection is a web application that supplies a modern and user-friendly interface for pizza businesses and their customers. This project will involve four team members each contributing to the development and deployment of this web application. Pizza Connection will provide customers of the pizza business with a clean and informative experience when choosing and placing their orders. Pizza businesses will benefit from a modern and effective interface offered by this application, enabling them to utilize various features and tools, such as inventory management, employee management, time tracking, and sales tracking. Overall, this project aims to bring about an application that will be efficient and easy to use for users and bring useful features and tools to one place so pizza businesses can make it part of their assets to aid their business and for their customers to be enticed by the easy and informative layout to purchase their pizza products.

## **1.2 Project Purpose, Scope, Objective**

## Purpose:

Today, it is tough to find a pizza app that works well for both customers and pizza businesses. Some pizza companies have their own websites for managing employees and inventory, but these websites are often too complicated and outdated. After looking at these websites closely, we have found some big problems we need to fix at once.

Scope:

This application will allow businesses to manage their employees and customers together.

The scopes of the project are:

1. Pizza Menu

2. Order cart

3. Order Payment

4. Employee Create

5. Manage Employee

6. Manage Inventory

7. Employee Time Clock

8. Manage Time Sheet

9. Order List

10. Contact Us

11. About us

Out-of-scope items:

1.Table reservations

2. Customer loyalty programs

3. Customer **s**upport,

4. Multi-language support.

Objective:

Our objective is to connect pizza sellers and customers directly, without using third-party sites or applications, and to provide an application with an easy and effective interface and toolset for pizza businesses to manage their business easily and efficiently. This application will achieve this by being easy for anyone to use, whether you're a customer or a pizza shop owner. It is user-friendly, so anyone can use it without any trouble.

## **1.3 Team Organization (Roles and Responsibilities)**

Team Lead - Matthew Krol,

UI Design Lead - William Esparza,

Technology Lead - Mehedi Zihad,

Documentation Lead - Kanta Islam

Team Lead

* Oversee team’s activities and efforts
* Team’s point of contact with client and GTA
* Assign task to team members
* Final says in debates

UI Design Lead

* Final says in user interface design
* Ensure design is meeting standards
* Designing and maintaining style guide

Technology Lead

* Ensures the correct technology for the project is used
* Guide for setting up among team members
* Assist to the best of ability in operations of the technology
* Any issues with technology will be tasked with answering

Presentation / Documentation Leads

* Documentation/presentation formatting
* Appearance, format and grammar
* Completeness of the document
* Accuracy of the document

Other Responsibilities

Each team member will contribute to completing each document and make any amendments as necessary for each step of the process. Each member will be contributing to the programming of the application from database setup, backend, frontend, and work to address any issues that need to be looked at. Testing and QA will be done by each team member. Any additional roles that may appear during the application's development will be discussed amongst the team members and the method of approaching these responsibilities or tasks will be addressed until then. The team will meet with the client (GTA). The team will be conducting its team meetings on Mondays, Wednesdays, and Saturdays.

## **1.4 Problem Resolution Policies**

As previously stated, due to the client's technical ability and the fact that this project builds upon an already existing software package the client will always be consulted first on the use of any non-prescribed technology and will be the deciding vote. On any intragroup decisions that do not involve the technical aspects of the project consensus of the group will be the deciding factor in the cases where this consensus cannot be achieved the group's GTA will be consulted, and the ultimate decision will exist with the team lead.

In the event a team member misses a scheduled meeting or a client-mandated deadline for delivery without notifying the group 48 hours (about 2 days) in advance the GTA and Professors will be notified of the event. Any further violations of this nature will result in a meeting with the Professors and GTA to decide the best course of action going forward. The habitual (more than once) missing meetings or non-client mandated deadlines will result in a verbal warning followed by the prescribed steps above.

In case the team leader does not attend meetings or meet deadlines the same procedure will happen above it will just be another member of the team that is handing out the written slips. In addition, everyone will act professionally and treat each other with respect. If this does not happen, the GTA will be made aware by one of the team members. This is to ensure that everyone is doing their role respectfully.

## **1.5 Project Plan (iterations, project schedule)**

Project Schedule:

Meeting with Client – 9/16/23

* Get answers to all setup questions.

Starting Website – 9/16/23

* Creating the login page
* Setting up a database for employees

First Prototype – 9/24/23

* Present the first prototype to the class.
* Display feature Home page
* Display feature Log-In
* Display feature Sign-Up
* Display feature Forgot Password.

Adding Features – 9/27/23

* Start working on features after completion of the basic website.
* Assign team members to work on features.

Testing Application – 10/12/23

* Vigorous test application to handle traffic.
* Vigorous test features to work correctly.

Second Prototype – 10/24/23

* Present second prototype to the class
* Employee Editing
* Display feature Time Clock
* Display feature Management

Enhancing Features – 10/27/23

* Enhance features from the demonstration.
* Start new features to complete the project.

Third Prototype – 11/12/23

* Present the third prototype to class.
* Highlighting updating features.
* Display feature Menu
* Display feature Inventory
* Display feature Payment
* Display feature Order List

Finishing Application – 11/15/23

* Work on features that have not been completed.
* Fixing any bugs or feedback from the third prototype
* Enhancing features if necessary

Final Presentation – 12/5/23

* Present working application

Team Meeting Schedule:

Monday & Wednesday (8:00 PM - 9:00 PM)

* Platform: Microsoft Teams (Online)

Saturday (5:00 PM - 7:00 PM)

* Location: Student Center, Wayne State University (In Person)

In addition to our scheduled meetings, we also have a WhatsApp group for issue resolution and utilize Microsoft Teams for quick or previously scheduled meetings without the need for additional links.

Client or GTA Meeting Schedule:

Tuesday (7:00 PM - 8:00 PM)

* Location: Room 0159, Old Main Building, Wayne State University (In Person)

## **1.6 Configuration Management Plan**

VCS Technologies:We use GitHub for our version control system (VCS). Our structure includes a main branch, a development branch, and individual branches for each team member.

Team members work independently on their features in their own branches, which originate from the development branch. Completed features are merged into the development branch following a thorough code review process.

The main branch contains stable and tested code, with the development branch merged into it after comprehensive testing and quality assurance checks.

We utilize GitHub’s collaboration features like issue tracking and project boards for task management. GitHub Actions are used for continuous integration (CI) and continuous deployment (CD), automating our testing and deployment processes.

Communication Channels: Our primary methods of communication method are WhatsApp and Microsoft Teams. Both platforms are widely used and offer convenient communication features. They have been mutually agreed upon by the team. We will rely on MS Word and MS PowerPoint for collaborative document creation and presentations. We have shared a folder in OneDrive for our pizza app documentation. These tools are chosen based on their familiarity with the team members and their availability to everyone. In addition, they offer simplicity, enabling us to create professional content efficiently without many hurdles.

The development plan for this project is to get the work done as a team. This includes tracking and controlling the changes to the project. To keep the project on track, we will have strict guidelines for when parts can be completed. Furthermore, as a team, we will be documenting any changes or new implementations that have been approved to enhance the

project. There will be reviews conducted by the team lead to ensure the project is kept on schedule.

## **1.7 Technologies**

For the pizza connection app, we have chosen the MERN technology stack at the recommendation of GTA and other group members.

The MERN stack is a popular web development framework made up of the stack of MongoDB,

Express.js, React.js, and Nodejs. It is one of the several variants of the MERN stack. MERN

adopts a full-stack development strategy, which requires the development of both frontend and backend components of an application. It enables quicker and more cost-effective product releases compared to other technology stacks.

Presentation Layer: ReactJS (Version: 18.2.0): We have chosen ReactJS (version 18.2.0) as the primary frontend framework for developing the web application and managing its visual aspects. Some of our team possesses experience in ReactJS, which will contribute significantly to creating a professional-grade product. By using ReactJS, we can enhance the responsiveness and usability of the UI components. To set up a solid foundation, we will utilize HTML5, CSS3, and JavaScript ES6 as the foundational languages, with ReactJS as a robust framework built upon them. The inclusion of HTML, CSS, and JavaScript is crucial as they are fundamental to web application development, playing indispensable roles in building a professional website.

Application Layer: ExpressJS & NodeJS (Version: 18.17.1): To implement the application layer we will use NodeJS with ExpressJS web framework. NodeJS is a JavaScript runtime environment for running JavaScript programs and is used to build server-side applications. ExpressJS is a web framework that enables us to design our Pizza Connection application to handle a variety of different HTTP demands.

Database Layer: MongoDB (Version: 6.0.7): For our database we use MongoDB. MongoDB is a NoSQL database in which each record consists of key-value pairs (Similar to JSON). It is a fast and document-oriented database that can easily index documents. It gives a faster response than other databases.