❖ Introduction

- **Explanation of differences (here you can explain issues with technologies along the way (i.e., lessons learned) or, perhaps, strategies that would better serve someone working on this project in the future**

- During the process of developing a drone solution for pharmaceutical distribution, a number of issues and challenges appeared, providing valuable information for other efforts in this area. One major problem was that there was no GPS built into the drone itself, which made navigating more challenging and required creative solutions like entering places as steps instead of normal addresses. Moreover, the focus was placed on the simplicity of the interface, ensuring that non-technical users, like temporary employees, could also easily utilize it.

❖ Technical Documentation

- **Software Design**

- The main focus of the software design was developing a Java user interface (UI) for entering addresses and managing the drone's movement. Flight management features of the Tello Drone were made possible by the backend's interaction with its API. To ensure a clear understanding of the drone's behavior, a state diagram was utilized to represent the drone's states during operation.

- **Tools used**

- The Java Development Kit (JDK) was used by the project to compile and execute Java code, and it was utilized to interface with the drone and control its flight. For development, an IDE like Eclipse was used, as it offers all the necessary tools for creating, altering, and troubleshooting code. Diagrams like class diagrams, state diagrams, and wireframes have been created made using

Unified Modeling Language (UML) tools to help visualize the project's architecture and design. Furthermore, Wi-Fi connectivity made it easier for the software interface and the Tello Drone to communicate, allowing for the smooth transfer of commands and status updates for effective drug delivery.

- **Dependencies/Assumptions**

- The Tello Drone's capabilities and the accessibility of its API for integration were critical components of the project. Regarding the drone's lack of GPS and the requirement for manual coordinate entry, assumptions were made. Furthermore, beliefs were made regarding the operating system, giving Windows OS compatibility first priority—a commonplace in pharmaceuticals retail environments.

❖ Evaluation

- **Test Plan**

- The purpose of the test plan was to guarantee the dependability and functionality of important components such flight routing, log history, and responsive user interface. Tests were created to mimic a number of scenarios, such as emergency situations, handling errors, and successful delivery.

- **Testing conducted**

- Project team members performed both manual and, where appropriate, automated testing during the testing process. To find errors, guarantee correct operation, and confirm the user experience, every feature was put through an intense testing procedure.

- **Results of testing**

- Testing produced insightful information about the software's functionality and performance. Iterative development cycles were utilized to fix identified

difficulties, resulting in a final product that is both robust and reliable.

❖ Future work

- **Where can this project go in the future?**

- There is a lot of room for improvement and extension with this project. Upcoming versions may include advanced capabilities like obstacle recognition, live drone status monitoring, and order management interface with backend services.

- **What are the next steps for your project if you were to continue working on it?**

- The following would need to be done if the project went forward: fixing any defects or usability issues that persisted, increasing productivity, and enhancing the current features in light of user feedback. Examining integration opportunities with delivery logistics platforms or other pharmacy management systems may potentially add value and utility to the project.