# **GOLGIX MANUFACTURING INTELLIGENCE PLATFORM DEPLOYMENT PLAN**

## Phase 1: Pre-Deployment Preparation

### 1. Define Objectives and Requirements

**Product Name: Golgix Manufacturing Intelligence Platform**

**Objectives:**

* **Optimize Customer Experience & Engineering:** The primary goal is to deliver an exceptional customer experience and provide robust engineering solutions. This involves leveraging cutting-edge technology to enhance user interaction and satisfaction. By catering to the specific needs of both customers and engineering stakeholders, the aim is to offer user-centric solutions and ensure seamless, effective functionality for both user groups.
* **Anticipate Machine and Process Failures with Advanced Notice:** One of the key objectives is to foresee potential machine and process failures well in advance. This predictive capability allows for proactive measures to be taken, ensuring sufficient lead time for necessary adjustments. By employing predictive models and analysis, the goal is to detect and mitigate potential issues before they escalate, thereby minimizing the impact on operations.
* **Reduce Unplanned Downtime and Production Losses:** A core objective is to significantly decrease unplanned downtime within the operational environment. By accurately forecasting potential failure points, the aim is to prevent and minimize unscheduled downtime. This proactive approach contributes to reducing production losses and maintaining a more consistent workflow, ultimately leading to enhanced productivity and resource optimization.
* **Enhance Plant Performance, Efficiency, and Profitability:** The overarching objective is to drive comprehensive improvement in plant performance, operational efficiency, and overall profitability. By fine-tuning operations based on predictive insights, the goal is to optimize various aspects of plant performance, leading to increased efficiency in processes and a subsequent improvement in profitability. This holistic approach aims to create a more streamlined and profitable operational environment.

**Target Audience:**

* **Frontline Operators:** Frontline Operators are equipped with real-time operational foresight using the Golgix Manufacturing Intelligence Platform. They receive immediate insights and predictive information that aid in proactive decision-making within the operational environment. This real-time foresight empowers them to adjust operational parameters to prevent machinery failures and optimize production efficiency.
* **Engineering Users:** Engineering Users leverage the platform primarily for operational reporting and gaining in-depth insights. They have access to a comprehensive database that provides crucial information for operational reporting, enabling them to derive insights from predicted root causes and eradicate downtime reasons systematically. This access supports their strategic decision-making to enhance overall operational efficiency and address production-related challenges effectively.

**Expected Outcomes:**

* **Enhanced Visibility, Foresight, and Process Optimization:** The product endeavors to provide heightened visibility into the operational landscape, offering foresight and direction for a seamlessly executed extrusion process. Through real-time insights and predictive capabilities, the aim is to optimize the operational workflow, thereby fostering a more efficient and controlled extrusion process.
* **Real-time Visualization of Plant Performance:** By incorporating real-time plant performance visualization, the goal is to provide a dynamic and immediate view of the plant's operational status. This feature allows for instantaneous monitoring and analysis, empowering quick decision-making and swift responses to any deviations or issues in the production process.
* **Advance Prediction of High-Pressure Failures:** A key focus is on accurately predicting high-pressure failures well in advance. This proactive approach enables operators to anticipate potential failures and take preventive measures, thus mitigating risks and ensuring uninterrupted operations.
* **Proactive Parameter Adjustments for Failure Prevention:** The project aims to facilitate proactive adjustments in operational parameters to prevent failures. This allows for dynamic and timely modifications to operating conditions, such as back pressure, temperature, and screw speed, to avert potential failures and extend operational efficiency.
* **Increased Productivity and Operational Empowerment:** The project strives to yield increased productivity within the operational environment. Empowering users with decision support tools and insights, the objective is to foster a more efficient and informed workforce, thereby enabling them to hit production targets and always know the likelihood of hitting them in real-time.

### 2. Establish a Cross-Functional Team

**Deployment Team:**

* **Product Owner (Solution Architect, ETL & deployment):** Serving as the Solution Architect, the Product Owner leads the deployment process, overseeing the design and implementation of the solution architecture. Responsible for Extract, Transform, Load (ETL) processes, and ensures the successful deployment of the platform.
* **Data Scientist (Automate Data Cleaning & ML models):** Focused on automating data cleaning processes and developing machine learning models for the platform.
* **Front-end Developer (User interface, features & functionality):** Responsible for crafting the user interface, implementing features, and ensuring functionality aligns with user requirements.
* **Back-end Developer (Containerization, Database & product features & functionality):** Responsible for containerization, managing the database, and implementing core product features and functionality.

**Specific Roles or Departments:**

* **Software Development Team:** Comprising roles responsible for the development and implementation of the software product, ensuring high standards and technical execution.
* **Client Success Department:** Engaged in post-deployment activities, ensuring client satisfaction, user support, and overall success of the deployed solution. They act as the bridge between the solution and end-users for a seamless user experience.

### 3. Deployment Strategy

**Deployment Strategy:**

* **Deployment Approach:** The strategy adopted for deployment is a **Ramped Deployment Strategy**, allowing for a phased and systematic implementation.

**Timeline and Milestones:**

Ramped deployment is a gradual and iterative approach designed to minimize risk and ensure a smooth transition from development to production. The strategy involves a step-by-step progression through various stages, gradually increasing the scope and scale of deployment while continually assessing and refining the solution.

Here's a breakdown of how the ramped deployment strategy will be executed:

1. **Initial Data Acquisition and Preparation (1 Week) :**

* **Stage Description:** Establishing the foundation by acquiring necessary data, treating it for optimal usage, and provisioning the required virtual machine infrastructure.
* **Execution Plan:** Initiate data access, extract, clean, and structure the data for training purposes. Provision the virtual machine and conduct user interviews to gather crucial insights.

1. **Model Training and Configuration (2 Weeks):**

* **Stage Description:** Developing and fine-tuning the machine learning model, configuring the user interface, and designing the deployment architecture.
* **Execution Plan:** Train the model, configure the UI, automate data transformation, evaluate model performance, and outline the deployment architecture.

1. **Comprehensive Technical Testing (2 Weeks):**

* **Stage Description:** Testing the model on real-time data, validating its accuracy, and performing deployment and UI/UX testing for quality assurance.
* **Execution Plan:** Conduct model testing on real-time data, execute deployment and UI/UX testing to ensure functionality and usability, and refine where necessary.

1. **Final Preparation and Testing (1 Week):**

* **Stage Description:** Finalizing the deployment package, launching the testing environment, and setting up the User Acceptance Testing plan.
* **Execution Plan:** Package the deployment solution, create the testing environment, and establish the plan for User Acceptance Testing (UAT) procedures.

1. **Deployment and User Feedback Collection (1 Week):**

* **Stage Description:** Deploying the finalized product on-premises, executing UAT, and collecting user feedback for improvements.
* **Execution Plan:** Deploy the final solution on-premises, conduct UAT sessions, and gather user feedback for further enhancements and refinements.

1. **Finalization and Official Launch (1 Week):**

* **Stage Description:** Customizing the user interface, deploying the finalized package, issuing login credentials, and officially launching the solution.
* **Execution Plan:** Customize the UI as per feedback, deploy the final product package, issue credentials, and execute the official launch of the solution.

### 4. Testing and Quality Assurance

**Quality Assurance and Feedback Collection:**

* **Testing Procedures:** User Acceptance Testing (UAT) phase involves assessing the solution's functionality, usability, and compliance with user requirements. During UAT, the system will be tested by actual users to ensure it meets business needs and functions correctly in a real-world environment.
* **Feedback Collection Mechanism:** In-Person Support for User Feedback by offering in-person and on-site support for users to provide direct feedback. Being on-site for the first-week post go-live will ensure their feedback is directly communicated and addressed, allowing for real-time discussions to gather comprehensive input and understand user needs and challenges accurately.

## Phase 2: Deployment Execution

### Pre-Launch Preparations:

**Infrastructure Readiness Checklist Overview:**

* **Hardware Requirements:**
  + Minimum 8GB RAM.
  + Approximately 100+ GB hard disk space for accommodating database growth.
  + At least 4 CPU cores. (not necessarily 100% dedicated)
* **Operating System:**
  + Linux-based OS, specifically Ubuntu or similar Debian-based Linux server distros.
* **Software Requirements:**
  + Support for Docker and Docker Compose on the virtual machine.
  + Ensure Git installation for initial deployment and updates.
  + Internet access to fetch data from external sources like GitHub for deployment.
* **Security Considerations:**
  + Avoiding exposure of external ports to the internet.
  + Restricting access to the web application outside the company intranet.
  + Ensuring a user with sudo (admin) privileges is available for deployment.
  + Providing command-line access such as SSH or VPN for deployment.
  + Enabling data upload without using externally exposed ports while initiating data uploads within the virtual machine.

**Contingency Plan for Deployment:**

* **Risk Identification and Analysis:**
  + **Identify Potential Risks:** Assess potential risks that might occur during deployment, such as hardware failures, network issues, or unexpected software conflicts.
  + **Risk Analysis**: Prioritize risks based on their potential impact on deployment and devise strategies to mitigate them.
* **Backup and Rollback Strategies:**
  + **Data Backups**: Regularly back up critical data, configurations, and system images to prevent data loss in case of unexpected issues.
  + **Rollback Procedures**: Prepare a documented rollback plan outlining steps to revert to the previous stable version if the deployment encounters significant issues.
* **Redundancy and Failover Measures (Client):**
  + **Hardware Redundancy:** Consider redundant hardware resources to ensure failover in case of hardware failure.
  + **Network Redundancy:** Ensure network redundancy and backup connections to maintain continuity in case of network failure.
* **Testing and Staging Environment:**
  + **Staging Environment:** Develop a staging environment to simulate the deployment process. Test the deployment in this environment to anticipate and address issues beforehand.
* **Monitoring and Alerts:**
  + **Monitoring Tools:** Implement monitoring tools to track the deployment process in real time. Set up alerts to notify the team in case of deviations or anomalies.
* **Regular Review and Updates:**
  + **Regular Review:** Schedule periodic reviews to assess the effectiveness of the contingency plan and update it according to evolving requirements.

### 2. Execute the Deployment

**Deployment Steps:**

**Data Extraction for Model Training:**

* + Acquire relevant data for training the predictive models.
  + Ensure data cleanliness and preparation for training.
* **Pellet Tech**
  + Input Data

**TIME-SERIES DATA (6 Months)**: Process Data from 4 Extruders (~ 1 min sampling)

* + - Extruder Pressure (psi)
    - Screw Speed Output (RPM)
    - Screw Speed (%)
    - Extruder Die Temp (deg F)
    - Extruder Thrust (%)
    - Feed Screw Current (Amps)
    - Discharge Conveyor Current (Amps)
    - Discharge Conveyor Speed (%)

**MANUALLY LOGGED DATA**

* Daily Production Spreadsheet (per 12hr shift)
* Equipment Failure log (per event)
* **User Interviews for UI Customization:**
  + Engage with end-users to gather insights for refining the user interface.
  + Incorporate feedback to enhance user experience and functionality.
* **Initial Product Package Deployment:**
  + Deploy foundational components of the product, including database, front-end, and essential functionalities.
  + Initiate the framework for subsequent development phases.
* **Comprehensive Model Training:**
  + Conduct thorough training for predictive models using acquired data.
  + Ensure model accuracy and readiness for real-time predictions.
* **Predictive Outcome Testing and Iteration:**
  + Test predictive models to assess accuracy and effectiveness.
  + Refine and iterate upon predictions for enhanced performance.
* **Database Schema Review and Enhancement:**
  + Evaluate the database structure for optimal efficiency.
  + Implement necessary alterations based on review findings.
* **Front-end Feature and Custom Screen Testing:**
  + Test user interface elements and custom screens for functionality and usability.
  + Refine and fine-tune interface components as needed.
* **Back-end Feature and Functionality Testing:**
  + Evaluate the back-end system to ensure optimal performance and reliability.
  + Address any functional discrepancies or performance issues identified.
* **Tablet Application Testing:**
  + Test the application on tablets to ensure compatibility and responsiveness.
  + Optimize the user experience for tablet users.
* **Notifications and User Profile Testing:**
  + Verify the functionality and effectiveness of notifications and user profile features.
  + Ensure a seamless user experience for interaction and engagement.
* **Final Product Package Deployment:**
  + Deploy the fully refined and optimized product package.
  + Prepare for the User Acceptance Testing (UAT) phase.
* **User Acceptance Testing (UAT) Execution:**
  + Conduct comprehensive testing with end-users to validate product readiness.
  + Gather final user feedback and address any outstanding issues.
* **Official Go-Live:**
  + Launch the product officially for operational use and end-user access.

## Phase 3: Post-Deployment Activities

### 1. Monitor Performance

**Automated Tracking for Key Metrics:**

* **Prediction Accuracy Tracking:** Automatic monitoring and notification system to track the accuracy of failure predictions daily. This ensures continual assessment and reporting of the accuracy of predictive models, enabling timely interventions and adjustments.
* **Downtime Aversion Validation:** Automated validation process to analyze and report instances where predictions averted potential downtime against cases where the system predicted downtime but it was not addressed by operators. Provides insights into the effectiveness of predictive models in preventing operational disruptions, quantifying the effectiveness of interventions in averting downtime.
* **Data Loss Tracking:** Automated system to track, log, and notify instances of data loss within the system. Enables identification of data loss patterns or issues for timely resolution and system improvement, ensuring data integrity and system reliability.
* **User Utilization Monitoring:** Automated tracking mechanism to monitor user engagement and correct system utilization. Aims to understand how effectively and accurately users are employing the system, ensuring proper utilization, and identifying potential training needs or system adoption and enhancements.

**Comprehensive Monitoring and Management Tools for Deployed Solution:**

* **Visualization and Data Analytics:** Utilize Apache Superset for comprehensive data exploration, visualization, and real-time monitoring to assess machine predictions and system performance efficiently.
* **System Monitoring and Alerting:** Consider Prometheus for seamless integration with Apache Superset, enabling efficient metric collection and real-time system anomaly detection for prompt issue identification and resolution.
* **ELK Stack (Elasticsearch, Logstash, Kibana) for Log Monitoring:** ELK Stack, a combination of Elasticsearch, Logstash, and Kibana, provides an effective log monitoring and analytics platform. It Allows comprehensive log analysis, enabling quick issue resolution, system troubleshooting, and in-depth performance evaluation through centralized log management.
* **Machine Learning Model Evaluation:** MLflow or TensorFlow remain recommended tools for precise evaluation and management of the machine learning lifecycle. These specialized tools offer comprehensive capabilities for model assessment compared to visualization-centric platforms like Superset.
* **User Activity Tracking:** For user tracking and behavior analysis, reliable options such as Mixpanel and Google Analytics can be employed to gain insights into user engagement and system utilization.
* **Data Loss Prevention Solutions:** In addition to Symantec and Digital Guardian, consider exploring alternative DLP solutions like McAfee DLP, Forcepoint, or open-source options such as OpenDLP or MyDLP. Selection should be based on specific requirements and budget considerations for effective data loss prevention.

### Enhanced User Support and Feedback Management:

**Service Availability:** Providing customer support from 8 AM to 5 PM, five days a week for immediate assistance and issue resolution. Ensuring consistent and accessible support during regular working hours to address user queries, technical concerns, and system-related issues promptly.

**On-Site Support Visits:** Conducting on-site visits to directly engage with users and address their questions, concerns, and feedback in real time. Offering personalized support and immediate problem-solving, facilitating a deeper understanding of user needs, and ensuring rapid resolution of any on-site system-related inquiries.

**Schedule Client Success Check-Ins:** Conduct weekly check-ins with the Client Success team to review user feedback, system performance, and ongoing issues. Facilitating a consistent feedback loop, ensuring the understanding of user experiences, addressing concerns, and strategizing for continual improvement and user satisfaction.

### Iterate and Improve

**Quarterly Model Re-training or Monthly Updates:**

**Iterative Model Enhancement:** Plan for ongoing model re-training every quarter to align with evolving data trends and ensure the machine learning models remain relevant and effective.

**Responsive Monthly Updates:** Immediate updates and adjustments as required, triggered by analysis and feedback, to address specific exceptions or critical issues that demand immediate intervention.