Industrial Internship Report on

”Prediction of Agriculture Crop Production in India”

Prepared by

[Divya Shree N]

Executive Summary
This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks' time. My project was (Tell about ur Project) This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface

Over the course of the past six weeks, our data science team has been diligently working on the prediction of agricultural crop production in India. Here's a summary of our work during this period: About need of relevant Internship in career development.

Data Collection and Preprocessing:

Exploratory Data Analysis (EDA):

Feature Engineering

Model Selection and Training:

Validation and Testing:

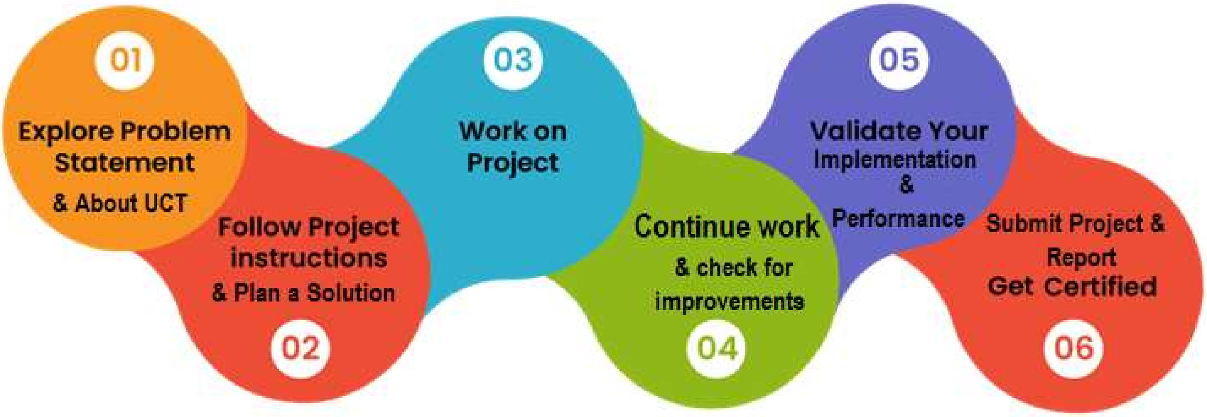
Model Interpretability:

2.Problem Statement

The project aims to predict agricultural crop production in India by leveraging data science techniques. The challenge lies in accurately forecasting crop yields amidst variable weather conditions, complex socio-economic factors, and regional disparities. The objective is to develop robust predictive models to support informed decision-making in the agriculture sector.

Opportunity given by USC/UCT.

How Program was planned



The project provided invaluable experience in data handling, model development, and collaboration, deepening my understanding of agriculture and predictive analytics.

Thank to all (UCT company and upskill team for providing these great opportunity and guiding us throughout the internship).

Your message to your juniors and peers.

To my juniors and peers, I encourage you to cultivate curiosity, never stop learning, and generously share your knowledge. Together, we can embark on a fulfilling journey of growth and discovery.

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

uct

Uniconverge Technologies

IIOT Products

We offer product ranging from Remote IOs, Wireless IOs, LoRaWAN Sensor Nodes/ Gateways, Signal converter and IoT gateways

IIOT Solutions

We offer solutions like OEE, Predictive Maintenance, LoRaWAN based Remote Monitoring, IoT Platform, Business Intelligence…

OEM Services

We offer solutions ranging from product design to final production we handle everything for you..

i. UCT IoT Platform (uct Insight)

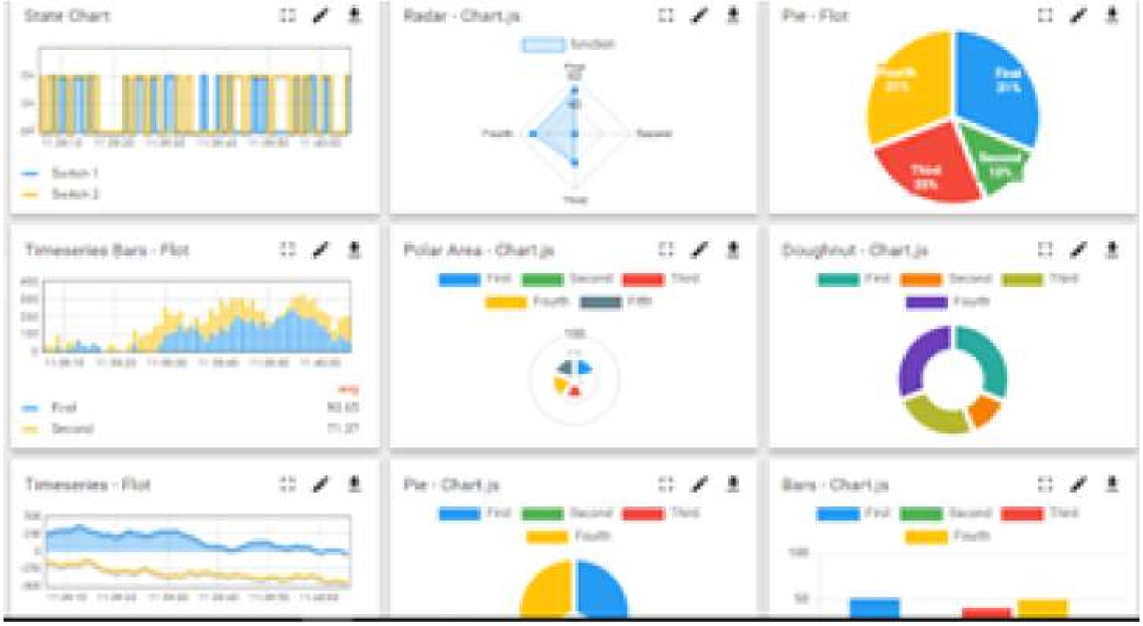
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

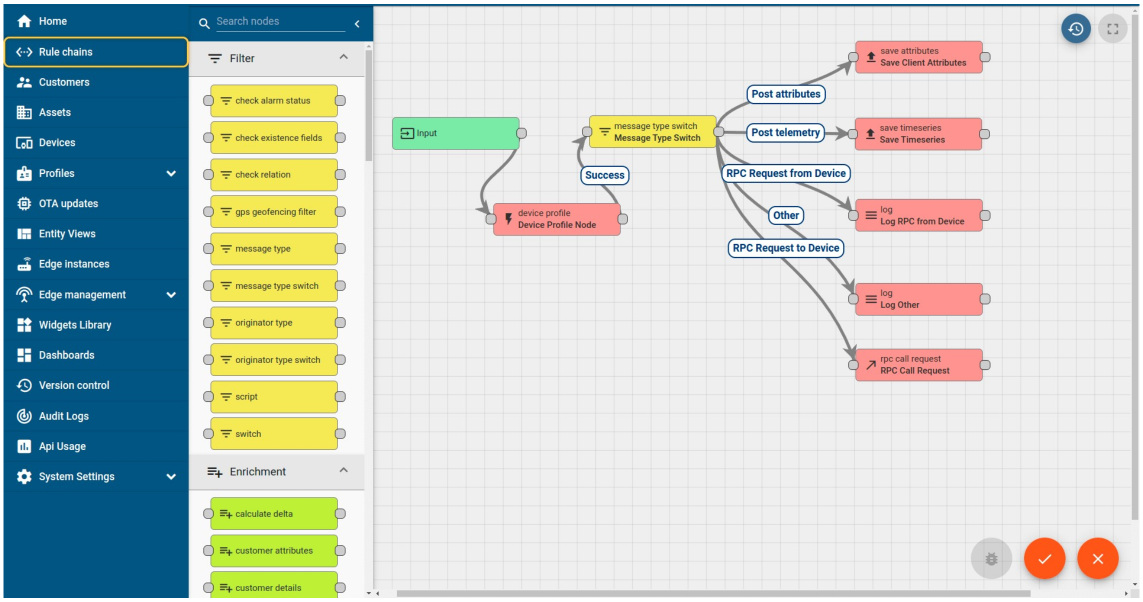
- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA

- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine





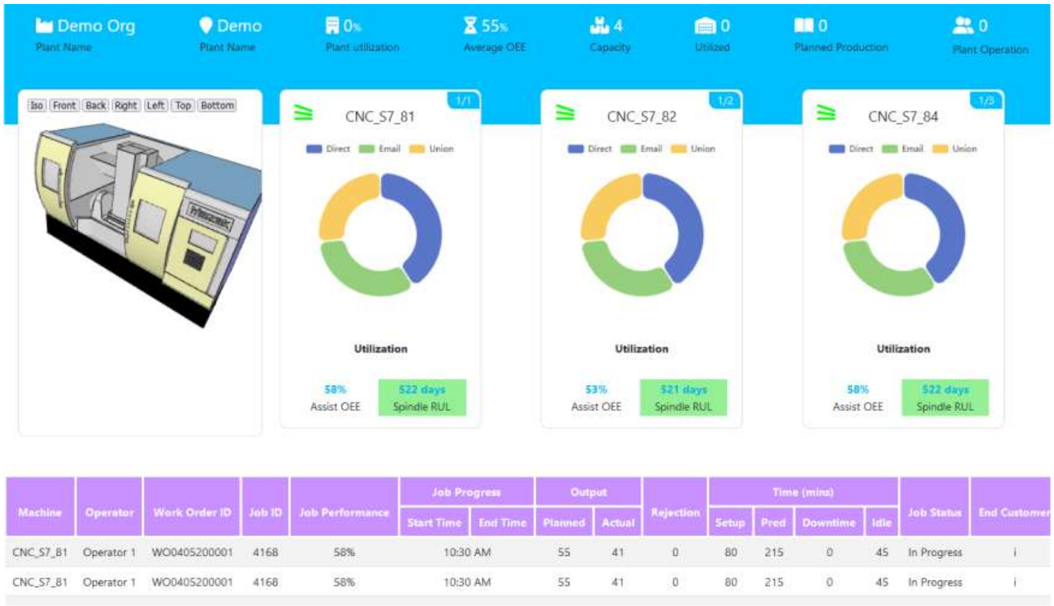
ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.

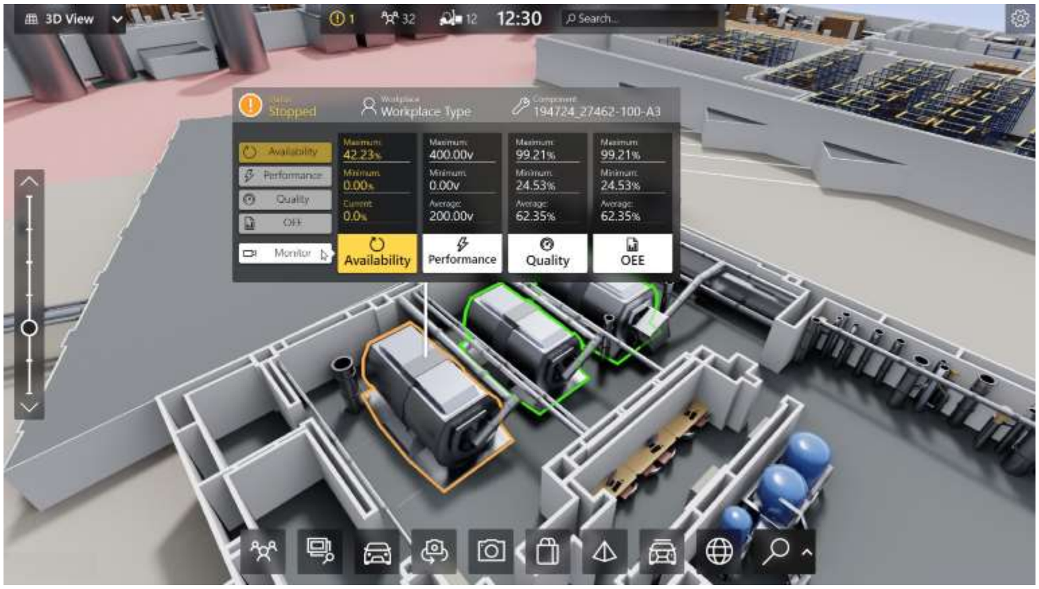
It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.



Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i

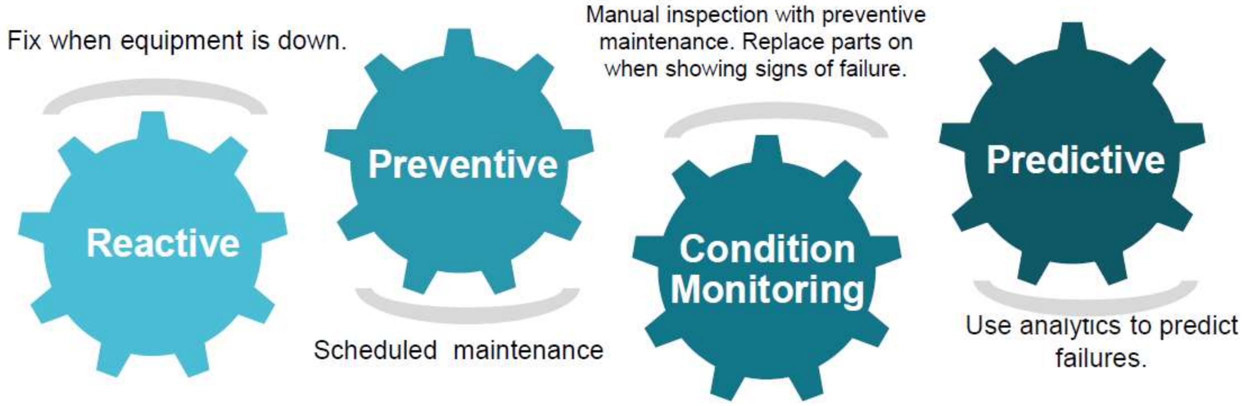


iii. LoRaWAN™ based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

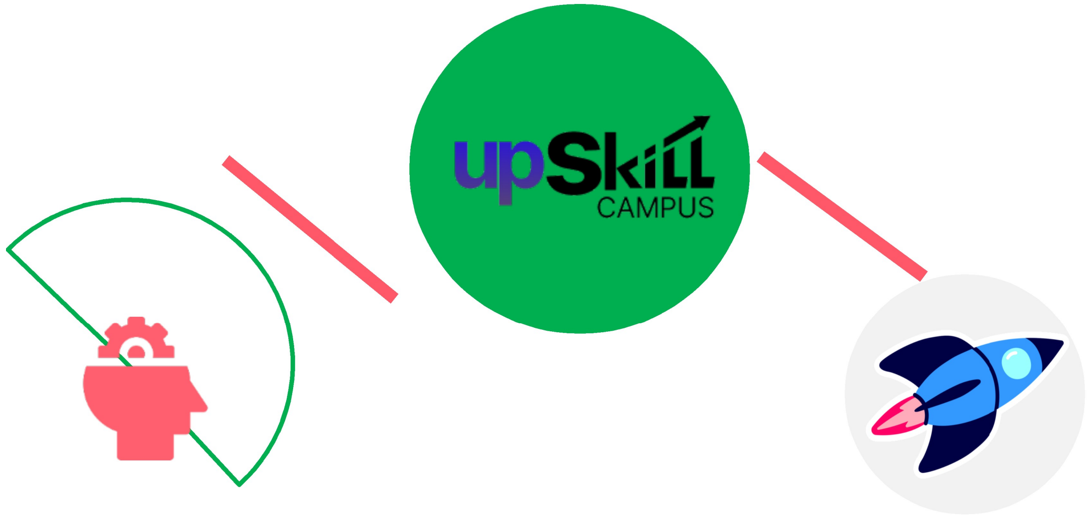
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.


Fix when equipment is down.
Reactive
Preventive
Scheduled maintenance
Manual inspection with preventive maintenance. Replace parts on when showing signs of failure.
Condition Monitoring
Predictive
Use analytics to predict failures.

2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

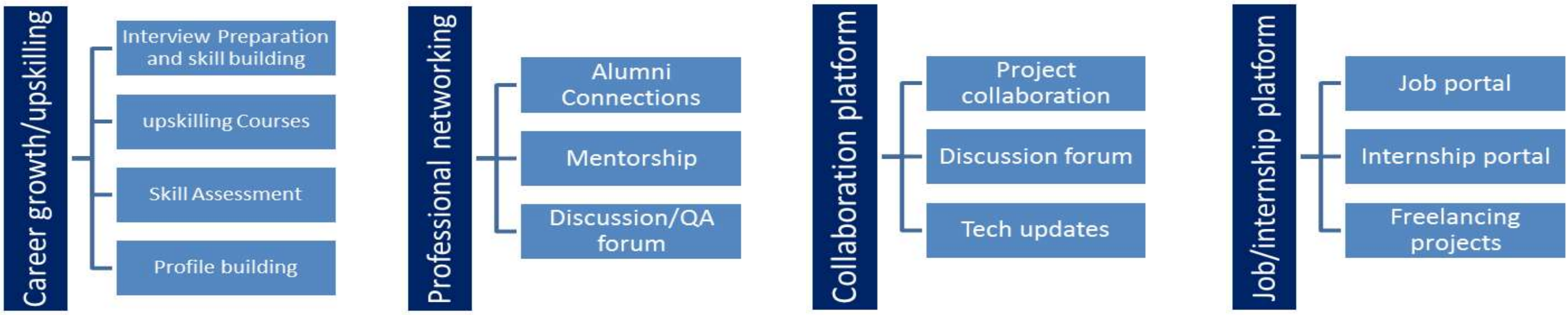
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



Career growth/upskilling
- Interview Preparation and skill building
- upskilling Courses
- Skill Assessment
- Profile building

Professional networking
- Alumni Connections
- Mentorship
- Discussion/QA forum

Collaboration platform
- Project collaboration
- Discussion forum
- Tech updates

Job/internship platform
- Job portal
- Internship portal
- Freelancing projects

2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4 Objectives of this Internship program

The objective for this internship program was to

☛ get practical experience of working in the industry.

☛ to solve real world problems.

☛ to have improved job prospects.

☛ to have Improved understanding of our field and its applications.

☛ to have Personal growth like better communication and problem solving.

2.5 Reference

1. Mentor or Senior Colleague

2. Academic Advisor or Professor

3. **Fellow Intern or Internship Program Coordinator**

3 Problem Statement

India's agriculture faces challenges due to climate variability, crop yield uncertainties, pests, water scarcity, and market price volatility. The goal is to provide farmers and policymakers with accurate and timely data-driven insights to optimize crop production, resource management, and sustainability. This entails leveraging technology, predictive models, and data integration while addressing the digital divide and promoting eco-friendly practices.

Agriculture in India grapples with climate variability, crop yield uncertainties, pests, water scarcity, market price volatility, and sustainability challenges. The objective is to empower farmers and policymakers with precise data-driven insights for crop optimization, resource management, and sustainability. This necessitates harnessing technology, predictive models, data integration, and addressing the digital divide. Additionally, it involves encouraging eco-friendly practices, providing education, and supporting access to technology for remote farmers.

4 Existing and Proposed solution

To address the multifaceted challenges of Indian agriculture, a holistic approach is required:

1. **Advanced Predictive Models:** Develop sophisticated predictive models using machine learning and artificial intelligence. These models should incorporate historical weather data, soil quality information, crop disease patterns, and market trends to provide accurate forecasts and recommendations.

2. **Weather Forecasting and Early Warning Systems:** Improve weather forecasting accuracy and establish early warning systems for extreme weather events. This information should be disseminated to farmers through mobile apps, SMS, and local outreach programs.

3. **Crop Health Monitoring:** Implement remote sensing technologies and satellite imagery to monitor crop health, detect diseases, and identify pest infestations. Provide real-time updates to farmers for timely interventions.

4. **Water Resource Management:** Promote precision irrigation techniques such as drip and sprinkler systems. Develop smart irrigation solutions that adjust water usage based on real-time weather and soil moisture data.

5. **Crop Diversification and Selection Tools:** Offer digital tools and advisory services to help farmers choose suitable crop varieties based on their specific conditions, including soil type, climate, and market demand.

6. **Market Price Information:** Create platforms or apps that offer real-time market price information, helping farmers make informed decisions on when and where to sell their produce for optimal prices.

7. **Sustainable Farming Practices:** Promote sustainable agricultural practices through training programs and incentives. Encourage crop rotation, organic farming, and the use of eco-friendly fertilizers and pesticides.

8. **Digital Access and Education:** Bridge the digital divide by ensuring farmers, especially those in remote areas, have access to smartphones and internet connectivity. Provide training and education on how to use digital tools effectively.

9. **Government Support and Policies:** Collaborate with government agencies to align policies and subsidies with the goal of enhancing agricultural sustainability and productivity. Advocate for investment in rural infrastructure and technology adoption.

10. **Data Integration and Platforms:** Develop a centralized data platform that integrates information from various sources, including weather stations, satellite imagery, soil testing, and historical agricultural data. Make this data accessible to farmers and researchers.

11. **Public-Private Partnerships:** Encourage collaboration between the government, private sector, research institutions, and NGOs to drive innovation, fund research, and support farmers' adoption of technology.

12. **Education and Extension Services:** Establish extension services that provide continuous support to farmers, disseminating knowledge on modern farming practices, pest management, and technological advancements.

13. **Research and Development:** Invest in research and development to continually improve predictive models, crop varieties, and farming techniques that are better suited to India's diverse agricultural landscape.

Implementing this comprehensive solution would require cooperation among various stakeholders, adequate funding, and ongoing evaluation to ensure its effectiveness in improving Indian agriculture's sustainability and productivity

Code submission (Github link): https://github.com/divya-shreen/upskillcampus.git

Report submission (Github link) :
https://github.com/divya-shreen/upskillcampus.git

4.1

5 Proposed Design/ Model

1. Data Collection and Integration:

- Gather data from multiple sources including weather stations, satellites, soil sensors, and historical agricultural records.
- Implement robust data integration to create a centralized repository.

2. Data Preprocessing:

- Cleanse and preprocess the data to remove noise and inconsistencies.
- Conduct feature engineering to extract relevant variables and transform data.

3. Predictive Models:

- Develop machine learning and AI models tailored to specific agricultural challenges:
  - Weather Forecasting Model: Utilize historical weather data and advanced forecasting techniques to predict weather patterns.
  - Crop Yield Prediction Model: Combine weather data, soil information, and crop-specific factors to forecast crop yields.
  - Disease and Pest Detection Model: Employ image recognition and data analytics to identify crop diseases and pests from images and sensor data.
  - Market Price Prediction Model: Utilize market data, demand-supply trends, and historical prices to predict market prices for agricultural products.

4. Model Training and Validation:

- Train the predictive models using historical data and fine-tune them for accuracy.
- Continuously validate models with real-time data to ensure reliability.

5. User Interface:

- Create an intuitive and accessible web or mobile application for farmers to easily access predictions, recommendations, and insights.
- Incorporate user feedback mechanisms for continuous improvement.

6. Alerting and Advisory System:

- Implement an alerting system that notifies farmers about critical weather events, disease outbreaks, or market fluctuations.
- Provide actionable recommendations based on predictions to guide farming decisions.

7. Data Security and Privacy:

- Ensure robust data security measures to protect sensitive agricultural data and farmer information.

8. Scalability and Accessibility:

- Design the system to be scalable, accommodating a growing user base and expanding data sources.
- Ensure accessibility for farmers in remote and rural areas, considering network connectivity challenges.

9. Collaboration and Stakeholder Engagement:

- Foster collaboration with government agencies, research institutions, and NGOs to share insights and best practices.
- Engage with farmers through training programs and workshops to maximize the system's benefits.

10. Continuous Improvement: - Establish a feedback loop for users to report issues and suggest enhancements. - Invest in research and development to stay at the forefront of agricultural technology.

This comprehensive Agriculture Prediction System integrates data analytics, machine learning, user-friendly interfaces, and stakeholder engagement to empower farmers, enhance crop productivity, and promote sustainable agriculture.

6 Performance Test

Performance testing is a crucial phase in software development and infrastructure management that evaluates the system's behavior and responsiveness under different conditions. Here are six key aspects and types of performance testing:

1. **Load Testing:** Load testing assesses how well a system performs under expected and peak loads. It helps identify performance bottlenecks, resource utilization, and response times when multiple users or transactions are accessing the system simultaneously.

2. **Stress Testing:** Stress testing pushes the system beyond its expected limits to determine its breaking point. This helps uncover vulnerabilities, such as memory leaks or unexpected crashes, when the system is under extreme load.

3. **Scalability Testing:** Scalability testing evaluates the system's ability to handle increased load by adding resources, such as servers or network bandwidth. It ensures that the system can efficiently scale to accommodate growing user demands.

4. **Performance Monitoring:** Continuous performance monitoring involves tracking system performance in real-time during its operational phase. It helps detect issues, optimize resource allocation, and ensure consistent performance over time.

5. **Response Time Testing:** Response time testing measures the time it takes for the system to respond to user inputs or requests. It focuses on ensuring that the system meets specific response time targets or service level agreements (SLAs).

6. **Capacity Planning:** Capacity planning is a proactive approach to performance testing. It involves forecasting future demands, estimating resource requirements, and ensuring that the infrastructure can handle anticipated loads to prevent performance issues before they occur.

Performance testing is essential for ensuring that software applications, websites, and IT infrastructure can handle user traffic, provide a smooth user experience, and meet performance expectations. It helps uncover and address performance bottlenecks, ensuring that systems are reliable, responsive, and scalable.

Test Plan/ Test Cases

Creating a comprehensive **test plan** and **test cases** is essential for ensuring the quality and functionality of a software application or system. Here's an overview of what a test plan is and how to structure it, followed by an example of a test case.

Test Plan:

A test plan is a detailed document that outlines the strategy, scope, objectives, resources, and schedule for testing a software application or system. It provides a roadmap for the testing process, ensuring that all aspects of the software are thoroughly examined and validated.

Components of a Test Plan:

1. **Introduction:**
   - Brief overview of the software or system being tested.
   - Purpose and objectives of the test plan.
2. **Scope and Objectives:**
   - Detailed description of what will be tested and what will not be tested.
   - Clear, measurable objectives for the testing effort.
3. **Test Strategy:**
   - Overview of the testing approach, including methodologies, techniques, and tools to be used.
   - Criteria for success and exit criteria for each testing phase.
4. **Test Schedule:**
   - Timeline with milestones, testing phases, and deadlines.
   - Resource allocation, including personnel, hardware, and software.
5. **Test Environment:**
   - Description of the testing environment, including hardware, software, and network configurations.
   - Dependencies and third-party integrations.
6. **Test Deliverables:**
   - List of documents and artifacts to be produced during testing, such as test cases, test data, and defect reports.
7. **Risks and Mitigation:**
   - Identification of potential risks that could impact the testing process.
   - Strategies and contingency plans for mitigating risks.
8. **Test Execution:**
   - Detailed procedures for executing test cases, including test data and expected outcomes.
   - Roles and responsibilities of testing team members.
9. **Defect Management:**
   - Procedures for logging, tracking, and prioritizing defects.
   - Defect resolution process and criteria for defect closure.
10. **Reporting and Metrics:**
    - Methods for reporting test progress and results.
    - Metrics to track testing effectiveness and quality.

Example Test Case:

Here's a simplified example of a test case for testing a login feature in a web application:

Test Case ID: TC001 **Test Case Title:** Login Functionality Test

Test Objective: To verify that users can successfully log in to the application using valid credentials.

Preconditions:

1. The web application is accessible and running.
2. A valid user account with the username and password is available for testing.

Test Steps:

1. Launch the web application.
2. Navigate to the login page.
3. Enter the valid username and password.
4. Click the "Login" button.
5. Verify that the user is redirected to the dashboard/homepage.
6. Check that the user's name or profile picture is displayed, indicating a successful login.

Expected Result: The user should be able to log in successfully, and the dashboard or homepage should load with the user's information displayed.

Postconditions: The user is logged in, and the application is ready for further testing.

By creating detailed test plans and test cases, you ensure that testing is systematic, thorough, and aligned with the project's objectives and requirements, leading to a higher-quality software product.

Test Procedure

1. **Test Case ID:** [Enter the unique identifier for the test case to which this procedure applies.]
2. **Test Case Title:** [Provide the title or description of the test case.]
3. **Objective:** [State the objective or goal of the test case.]
4. **Preconditions:** [List any prerequisites or conditions that must be met before executing this test case.]
5. **Inputs:** [Specify any input data, parameters, or conditions that need to be set up for the test.]
6. **Test Steps:**
   - **Step 1:** [Describe the first action or step to perform during the test. Include all necessary details such as what to click, where to navigate, or what to input.]
     - Action: [Specify the action or task to perform.]
     - Expected Result: [Clearly define the expected outcome or result of the action.]

- **Step 2:** [Continue with the next action or step. Repeat this format for each step in the procedure.]

7. **Cleanup Steps:** [List any actions or steps that should be taken to reset the system or environment to its initial state after the test.]
8. **Expected Results:** [Provide a summary of the expected results or outcomes of the entire test case.]
9. **Postconditions:** [Describe the system's state or conditions after the test has been executed.]
10. **Notes and Additional Information:** [Include any additional notes, details, or considerations relevant to the test procedure.]
11. **Attachments:** [Attach any relevant documents, screenshots, or files that may aid in executing the test.]

Example Test Procedure:

Here's a simplified example of a test procedure for the test case mentioned earlier, "Login Functionality Test":

Test Case ID: TC001 **Test Case Title:** Login Functionality Test

Objective: To verify that users can successfully log in to the application using valid credentials.

Preconditions:

1. The web application is accessible and running.
2. A valid user account with the username and password is available for testing.

Inputs: Valid username and password.

Test Steps:

- **Step 1:** Launch the web application.
  - Action: Open the web browser and enter the application's URL.
  - Expected Result: The application's login page is displayed.
- **Step 2:** Navigate to the login page.
  - Action: Click on the "Login" link or button.
  - Expected Result: The login form is visible.
- **Step 3:** Enter the valid username and password.
  - Action: Input the valid username and password into the respective fields.
  - Expected Result: The credentials are entered without errors.
- **Step 4:** Click the "Login" button.
  - Action: Click the "Login" button on the form.
  - Expected Result: The user is redirected to the dashboard or homepage.

- **Step 5:** Verify that the user is redirected to the dashboard/homepage.
  - Action: Observe the page that loads after clicking "Login."
  - Expected Result: The dashboard or homepage is displayed.
- **Step 6:** Check that the user's name or profile picture is displayed, indicating a successful login.
  - Action: Look for the user's name or profile picture on the dashboard.
  - Expected Result: The user's name or profile picture is visible, confirming a successful login.

Cleanup Steps: No cleanup steps are required for this test.

Expected Results: The user should be able to log in successfully, and the dashboard or homepage should load with the user's information displayed.

Postconditions: The user is logged in, and the application is ready for further testing.

By following a structured test procedure, testers can systematically execute test cases, record results, and ensure that testing is consistent and repeatable.

Performance Outcome

- **Skill Development:** Successful interns should acquire and improve various skills related to their field of study or the internship role. This can include technical skills, communication skills, problem-solving abilities, and industry-specific knowledge.

- **Project Contribution:** Interns are expected to contribute to ongoing projects or initiatives within the organization. This may involve completing tasks, delivering projects, or assisting in research and analysis.

- **Learning and Adaptation:** Interns should be proactive learners who quickly adapt to new environments, technologies, and processes. They should be able to apply classroom knowledge to real-world situations.

- **Teamwork and Collaboration:** Interns need to effectively collaborate with colleagues and supervisors. They should demonstrate the ability to work in a team, communicate clearly, and contribute positively to group dynamics.

- **Professionalism:** Professional conduct is vital for interns. This includes punctuality, reliability, respect for company policies, and maintaining a professional attitude in the workplace.

- **Problem Solving:** Interns are expected to identify and solve problems independently or in collaboration with their team. They should demonstrate critical thinking and creative problem-solving skills.

- **Communication Skills:** Clear and effective communication is essential. Interns should be able to convey their ideas, ask questions when needed, and provide updates on their progress.

- **Networking:** Building professional relationships and networking within the organization can be a valuable outcome.

My learnings

1. **Technical Skills:** Depending on the nature of the project, you may acquire or enhance technical skills relevant to your field of study or the internship role. For example, if you're in a software development internship, you might learn programming languages, database management, or software testing techniques.

2. **Problem-Solving:** Real-world projects often present complex challenges. Through your internship project, you can develop problem-solving skills by tackling issues, finding solutions, and making informed decisions.

3. **Project Management:** If your project involves coordinating tasks, meeting deadlines, and managing resources, you can learn project management skills. This includes task prioritization, time management, and project planning.

4. **Teamwork and Collaboration:** Working on a project typically involves collaboration with team members. You may learn how to communicate effectively, resolve conflicts, and contribute to a positive team dynamic.

5. **Adaptability:** Internship projects often require you to adapt to changing circumstances, such as evolving project requirements or unexpected obstacles. This adaptability is a valuable skill in any career.

6. **Research and Analysis:** You might gain experience in conducting research, gathering data, and analyzing information to support your project's objectives. This skill is transferable to various fields.

7. **Communication Skills:** Your project work can improve your ability to convey complex ideas clearly and concisely, whether through written reports, presentations, or discussions with colleagues.

8. **Attention to Detail:** Many projects demand a high level of precision and attention to detail. You may learn to spot errors, inconsistencies, or potential issues that could affect project outcomes.

9. **Critical Thinking:** Through problem-solving and decision-making, you can develop critical thinking skills. This involves analyzing situations, evaluating options, and making well-informed choices.

10. **Networking:** Working on a project allows you to connect with colleagues, mentors, and industry professionals. Networking can open up future opportunities and provide valuable insights.

11. **Professionalism:** An internship project can teach you the importance of professionalism, including workplace etiquette, punctuality, and meeting deadlines.

12. **Industry Knowledge:** If your project is industry-specific, you can gain a deeper understanding of the sector, its trends, and the challenges it faces.

13. **Self-Confidence:** Successfully completing a project can boost your self-confidence and provide a sense of accomplishment.

14. **Self-Management:** Managing your time and tasks effectively during a project can help you become more organized and improve your self-management skills.

15. **Feedback Acceptance:** Receiving feedback from colleagues or supervisors during the project can help you develop the ability to accept constructive criticism and use it for personal growth.

7 Future work scope

1. **Performance Metrics:** Determine if there is an opportunity to refine or expand the project's performance metrics. Consider incorporating more specific and measurable key performance indicators (KPIs) to track the project's success and impact.

2. **Scaling and Growth:** Explore whether the project can be scaled up to a larger scope or implemented in different areas or departments within the organization. Identify potential opportunities for expansion.

3. **Enhancements and Refinements:** Assess if there are features or functionalities that could be added or improved to enhance the project's effectiveness or user experience. Consider collecting user feedback for these improvements.

4. **Automation:** Investigate the potential for automating certain aspects of the project to reduce manual effort and increase efficiency. Automation can lead to cost savings and improved accuracy.

5. **Integration:** Explore opportunities to integrate the project with other systems or tools within the organization. This can enhance data sharing, streamline processes, and create a more cohesive ecosystem.

6. **Advanced Analytics:** Consider implementing more advanced data analytics or machine learning techniques to extract deeper insights from the project's data. This can lead to more sophisticated predictions or recommendations.

7. **User Training and Adoption:** Assess whether additional training or educational materials are needed to encourage wider user adoption of the project's outcomes. Training programs can help maximize the project's impact.

8. **Long-Term Sustainability:** Discuss strategies for maintaining the project's benefits over the long term. This might involve regular updates, data maintenance, or the establishment of standard operating procedures.

9. **Monitoring and Evaluation:** Develop a plan for ongoing monitoring and evaluation to ensure the project continues to meet its objectives and adapt to changing needs or circumstances.

10. **Collaboration Opportunities:** Explore opportunities for collaboration with other teams, departments, or external partners who may benefit from the project's outcomes or contribute to its success.

11. **Budget and Resource Allocation:** Determine the budget and resource requirements for future work on the project. This includes personnel, technology, and other resources needed for expansion or enhancements.

12. **Feedback Mechanisms:** Establish mechanisms for collecting feedback from end-users, stakeholders, and team members to continuously improve the project's performance and relevance.

13. **Documentation and Knowledge Transfer:** Ensure that project documentation is comprehensive and accessible to future team members or stakeholders who may be involved in the project's continuation.

14. **Risk Assessment:** Identify potential risks or challenges associated with future work on the project and develop mitigation strategies to address them.

15. **Project Roadmap:** Create a roadmap or plan that outlines the timeline and milestones for future work on the project, including clear objectives and deliverables.