Technical Specification Double-side Page

1. **TECHNICAL SCOPE:** Summarize the mock-up devised during the EXPLORE phase: how have you addressed the challenge/Theme Challenges and tackled with its requirements and data. Include a diagram.

DESCRIPTION

The solution is a service (SasS platform) to manage, analyse and visualise order compliance using computer vision. Wiibiq platform identifies the items that exist in images taken with in-place cameras during the picking process to verify the completeness of the orders.

A user logs into Wiibiq through a web browser to monitor the picking process, the order status, and the details of the items (medicines) detected during the picking process. The images collected are attached to each order with the corresponding detected classes and accuracy. Also, the solution includes a dashboard with real-time and historical information related to the picking operations as shown in Figure 1. In addition, a mobile application is used to visualise alerts and handle manual verifications when required.

HOW WE ADDRESS THE PROBLEM

We tackle the challenge using transfer learning and leveraging the recent advancements in computer vision as a complement of our SaaS solution for inventory and manufacturing management. The solution collects sample data with different medicine images. Then, we label these images. Finally, we train a model using the annotations. These models are used for inferences from the web application so that the user can exploit the generated data to monitor the picking process.

Figure 1. Wiibiq dashboard screenshot with metrics showing the status of current and previous orders.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981

SAAS PLATFORM ARCHITECTURE

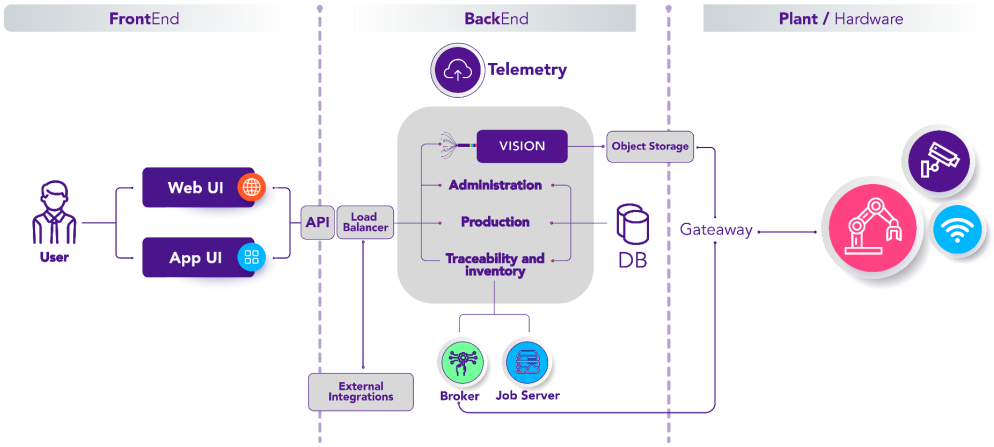

Figure 2. Wiibiq High Level Architecture.

2. **ALGORITHMS, TOOLS AND CONCLUSIONS:** Detail the algorithms and tools identified to accomplish the challenge/Theme Challenges. Show clear understanding of the used REACH dataset/s and addressed challenge/Theme Challenges.

AI TRAINING AND DETECTION

The Wiibiq Vision module carries out two procedures. The first one is an iterative process that starts with image collection and finishes with model deployment to production, as illustrated in Figure 3. The second process detects the items in an image. It sends the results to the web application to trigger alerts and display visual information. The steps associated with the mentioned processes are described below.

Model training and deployment process.

1. Collect images using Google Cloud Storage API and store them in a machine-readable format using Apache NiFi.
2. Hash the information related to the bucket ID to guarantee confidentiality.
3. Images are preprocessed according to the resolution, formats, sizes, and noise.
4. Label images (manually and/or using artificial intelligence labelling techniques).
5. Train and evaluate the detection models.
6. Evaluate the models with the following metrics: Intersection-over-Union (IoU), accuracy, precision, F1 score, and mAP.

Object detection process.

1. Collect images from Google Cloud Storage (provider data storage) using Apache NiFi.
2. Send images to the detection RESTful API.
3. Use Kafka to do the inference and send results, alerts and notifications according to preconfigured rules.
4. Display results in the web application.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981

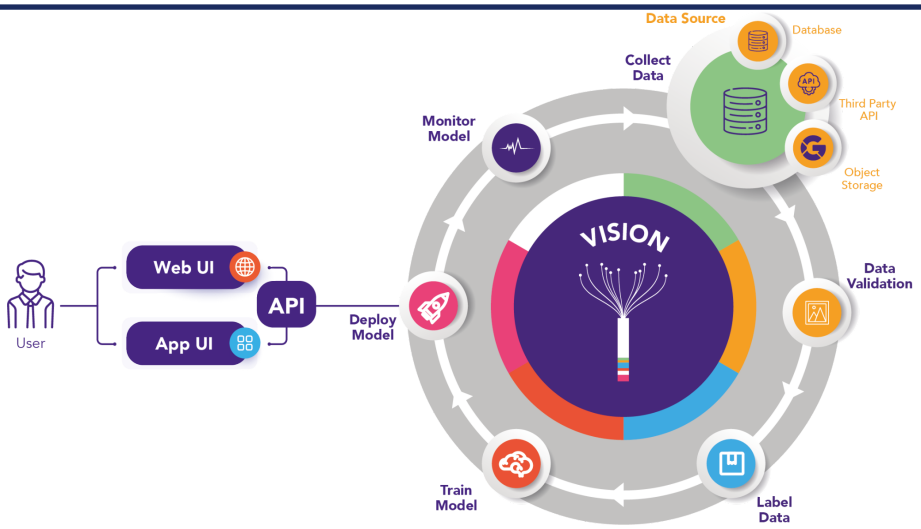

Figure 3. Data flow for solution deployment.

YOLOv5 ALGORITHM

We adapt the YOLOv5 pre-trained models to detect the items (medicines) by using the corresponding dataset and configuring hyperparameters to compare the achieved performance.

The YOLOv5 consists of the following main parts:

1. The model backbone (CSP - Cross Stage Partial Networks) extracts rich and useful characteristics from an input image.
2. The model neck is used to create feature pyramids to generalise better with different object sizes and scales. PANet is used as a neck in YOLO v5 to get feature pyramids.
3. The model head is primarily responsible for the final detection step. It uses anchor boxes to construct final output vectors with class probabilities, objectness scores, and bounding boxes.

TOOLS

The infrastructure provided by the University of Deusto facilitates the deployment of the solution using Kubernetes. We use containers to deploy our application and the related services such as API endpoints (Flask and Django). Also, we will use tools like TensorFlow and Pytorch for the training process and Jupyter Notebooks for experimentation.

3. **SCALABILITY AND FLEXIBILITY OF THE SOLUTION:** Discuss whether the solution can truly cope with humongous and increasing datasets and how flexible it is to adapt to other related domains

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981

SCALABILITY

We use Kubernetes to deploy our solution. The metrics server enables Kubernetes to create pipelines to scale and distribute the workload automatically to match demand. Also, we implement CI/CD pipelines to manage a workflow that is executed through independent modules. These modules collect, validate and process data; train, validate and test the models; deploy the solution and monitor the quality in production. Apache NiFi and Kafka are used to manage the flow of data in a secure and scalable way.

FLEXIBILITY

The system can be adapted to other domains using transfer learning. This method allows us to store the knowledge gained in solving a particular problem and transfer it to another field. We require a dataset with images related to the new problem to achieve it. Therefore, our model's experience will increase over time, facilitating the adaptation of the solution to solve other problems related to visual inspection, such as correct product labelling and foreign object detection. In addition, the solution can be adapted to different scenarios with bandwidth limitations, latency issues, and unpredictable network behaviours using edge computing and small versions of the model.

4. **DATA GOVERNANCE AND LEGAL COMPLIANCE:** Describe the security level of the proposed solution, i.e. how authentication, authorization policies, encryption or other approaches are used to keep data secure. Explain how will be compliant with the current data legislations concerning security and privacy (e.g. GDPR).

Wiibiq information security policies define processes and tools to manage data, resources, and permissions and prevent security incidences. We are continuously auditing the data collection processes to be able to comply with GDPR. We identify the minimum amount of personal data to collect and store. Also, data collected is under the control of data owners according to the service contracts.

Access to the Wiibiq platform through our web and mobile applications uses a 2FA (two-factor authentication) mechanism. We use role-based authorization to separate views and data according to user privileges. Also, an API Key enables access to the Wiibiq API only to authenticated users. In addition, we enforce SSL to restrict requests with client certificate authentication, and we use a load balancer to manage thousands of requests without affecting the system's availability.

In the development process, we avoid the use of any end-of-life software. Also, regular security audits allow us to verify the existence of potential vulnerabilities; moreover, software components are updated to patch vendor issues if vulnerabilities and exposures are disclosed.

We deploy the Wiibiq solution components in the cloud and monitor them with a telemetry system. Isolated microservices decrease the potential attack's surface area, while independent databases restrict access according to the domain and level of privileges. Telemetry allows us to control request rates, dependency rates, and exceptions and centralises the logs related to the different parts of the system. Finally, security processes are integrated as part of the CI/CD pipeline to perform checks and validations systematically and continuously.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981

5. **QUALITY ASSURANCE AND RISK MANAGEMENT:** Describe the quality process planed for the final product. Technologically, which are the potential risks in all the phases of the project (design of the solution, development, testing, deployment...) and indicate mitigation plans to still fulfil the challenge/Theme Challenges and data provider requirements.

The identified risks related to our solution include rapidly changing tools and frameworks, the use of hard-to-interpret metrics, complex model training, the adaptation of models for production, errors recovery, and speed of iteration. Therefore, to fulfil the challenge according to data provider requirements, we consider the following actions in our mitigation plan:

- Integrate incremental design to facilitate the testing and debugging processes during smaller iterations.
- The creation of automation pipelines to decrease errors when performing repetitive steps like the build, test, and deployment tasks.
- The creation of different sized datasets to train the models and test them in different periods.
- Monitor the development process to detect events impacting or altering the risk surface.
- QA processes that validates the interoperability of the system to guarantee successful integrations with external components.
- Integrate redundancy in the data infrastructure and perform regular backups to prevent data loss and hardware failure.
- Monitor the performance, security, and quality of the infrastructure and data sources using telemetry.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951981