Location-independent Strategy for the Work Environment of the Future

Project: Offenes Innovationslabor KI zur Förderung gemeinwohlorientierter KI-Anwendungen (Go-KI)

Authors: Pedram Babakhani, Finn Harms, Moritz Heine, Jan Henri Evard, Lennard Hänsch, Tobias Schulz, Tobias Küster, Fikret Sivrikaya

Date: 17.03.2025

Content

Gefördert durch:

Bundesministerium für Arbeit und Soziales

aufgrund eines Beschlusses des Deutschen Bundestages

Introduction

One goal of the *Go-KI* project was to investigate methods and tools for the low-threshold development of common-good AI applications with a focus on the workplace of the future. The *Center for Tangible Artificial Intelligence and Digitalization (ZEKI)*, located in Berlin, serves as an innovation hub focused on advancing applied artificial intelligence (AI) in tangible and interactive environments. As part of this initiative, an experimental digitised workspace, the ZEKI "*Real Laboratory*", was established as a dedicated sub-project within the Go-KI project,

The primary objective of the Real Laboratory was to create a flexible, experimental space that acts as a playground for *smart space* applications and interactive technologies. This space facilitates the design, testing, and evaluation of cutting-edge AI-driven solutions intended for future workplaces and interactive environments. Beyond experimentation, the environment provides a platform to collect, process, and analyze data, offering insights into the integration of AI technologies into daily life while promoting sustainability and user-centric design principles.

In this context, a number of different innovative applications have been developed and tested – in part in the course of the Ambient Assisted Living (AAL) project course by TU Berlin, and in part directly in the Go-KI project. Those applications span a wide range of use cases, from improved well-being in the office to productivity and sustainability in the workplace. Besides those individual applications, the Go-KI project also resulted in the development of an AI Framework and associated tools that can be used to interact with and orchestrate those applications.

This document provides an overview of the ZEKI Real Laboratory environment and the different applications that have been developed in that context. It describes each of the applications in detail, including what would have to be considered in order to transfer them to a similar environment. Finally, it also showcases the Go-KI AI Framework "OPACA" which can be used to combine and orchestrate many of the functions provided by those applications.

Literature Survey

Smart spaces represent a transformative advancement in integrating digital technologies with physical environments to enhance human interaction, productivity, and well-being. These environments are equipped with interconnected sensors, IoT devices, and artificial intelligence (AI) systems to enable adaptive and context-aware functionalities. The significance of smart spaces lies in their ability to bridge the digital and physical worlds, facilitating seamless interaction and enabling applications in areas like healthcare, education, and workplace optimization.

The exploration of experimental environments, such as living labs and intelligent spaces, has been a focal point in research aimed at advancing smart space applications. Living labs, as highlighted by Pan (2023), emphasize collaborative spaces for co-creation and innovation in smart cities. These labs integrate end-users into the development cycle, ensuring practical solutions tailored to real-world needs.

“Intelligent Space” environments, as described by Sasaki and Hashimoto (2007), employ distributed sensors and actuators to facilitate seamless human interaction and robotic collaboration. This work provides the foundation for integrating advanced AI systems into tangible environments.

Diraco et al. (2019) explored early change detection using AI in smart living environments, focusing on behavioral monitoring to enhance safety and comfort. Similarly, Rivera-Illingworth et al. (2006) demonstrated the use of neural networks to identify human activities in pervasive living spaces, contributing to the development of adaptive systems.

The role of multimodal interaction is emphasized by Bui and Chong (2018), who proposed an integrated human-robot-environment interaction framework for ambient-assisted living. Their work underscores the importance of intuitive and user-friendly interfaces in smart environments.

In the context of behavior recognition, Rodríguez and Natalia (2015) implemented fuzzy ontologies to model human activities, enabling smarter interactions in AI-driven spaces. Additionally, Braun et al. (2017) developed capacitive sensors for subtle human sensing in smart living spaces, addressing the need for non-invasive data collection technologies.

A practical infrastructure for studying activity monitoring needs in service robotics was introduced by Vasileiadis et al. (2016). This living lab framework highlights the value of testing AI applications in real-life scenarios.

These works collectively demonstrate the potential of experimental environments like the Real Laboratory in advancing smart space technologies. By building on these foundations, the Real Laboratory integrates state-of-the-art applications with a focus on adaptability, user-centric design, and practical implementation.

The ZEKI Environment

The Center for Tangible Artificial Intelligence and Digitalization (ZEKI, Zentrum für Erlebbare Künstliche Intelligenz und Digitalisierung) spans an 813 m^2 office space in Berlin, designed to support innovative research, development, and experimentation in artificial intelligence and digitalization. The layout of ZEKI emphasizes both functionality and collaboration, with carefully planned spaces tailored to diverse applications and interdisciplinary teamwork.

The floor plan illustrates the variety of dedicated rooms and their specific purposes. ZEKI's environment includes specialized spaces such as the Delivery Robot Room that is equipped for the testing and operation of autonomous robotic systems in the context of the BeIntelli[1] project. These areas provide a controlled setup for developing cutting-edge delivery technologies.

ZEKI also prioritizes collaborative and creative workflows. The Co-Working Space is a dynamic area designed for flexible teamwork, while the Design-Thinking Space is intended for creative brainstorming sessions. For focused, individual work, the Focus Space offers a quiet, distraction-free environment. Additionally, the Conference Room supports large meetings and presentations, featuring state-of-the-art audiovisual equipment to facilitate knowledge sharing and discussions. A mute-box can be used for phone calls and video conferences, without disturbance from and to the other colleagues in the open office.

FIGURE: Impressions from the ZEKI work environment, illustrating the variety of the rooms for the purposes of collaboration, demonstration, focused working, brainstorming, and socializing

[1] https://be-intelli.com/

A standout feature of ZEKI is the Experience Hub, which acts as a combined auditorium and showroom for AI-driven technologies. This interactive space allows visitors to engage directly with advanced solutions, providing an immersive introduction to the possibilities of artificial intelligence. Adjacent amenities such as the Smart Kitchen further enhance the usability of the space, ensuring a comfortable and practical work environment for employees and visitors alike.

The Real Laboratory at ZEKI is an innovative experimental platform that embodies the principles of smart spaces. It serves as a testing ground for AI-driven applications, including wayfinding systems, smart desks, and gesture-based controls. By providing a controlled environment for studying human interaction with advanced technologies, the Real Laboratory contributes to the development of user-friendly and efficient AI systems. Its unique approach combines tangible AI with real-world application testing to advance the design and functionality of intelligent environments.

The following figure provides an overview of the ZEKI layout, highlighting the seamless integration of functional, collaborative, and interactive spaces. This versatile environment exemplifies how cutting-edge technology can be embedded into modern workspaces to foster creativity, productivity, and innovation.

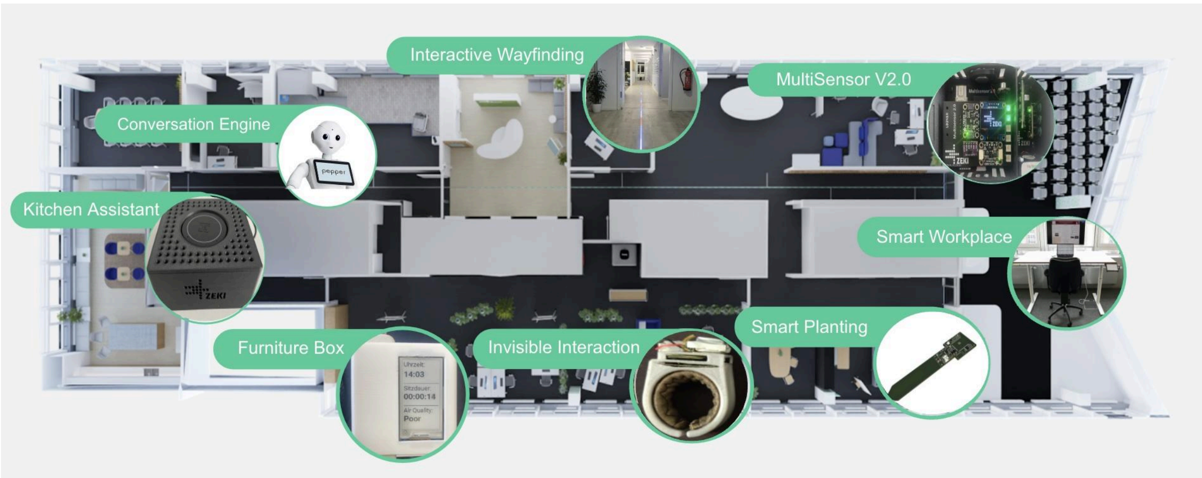

FIGURE: Floor plan of the ZEKI office, showcasing the layout and purpose of various rooms designed for innovative research and collaboration. Applications include the Kitchen Assistant, Interactive Wayfinding, Smart Planting, and Smart Workplace, showcasing the Real Laboratory's innovative exploration of smart environments.

Applications in the ZEKI Real Laboratory

The Real Laboratory serves as an experimental playground for exploring state-of-the-art smart space technologies. This space allows for real-world data collection, user interaction testing, and iterative development of AI-driven applications. Each of the applications implemented in the Real Laboratory is designed to explore innovative approaches to improving user experience, workplace efficiency, and environmental adaptability.

Besides providing individual user interfaces for their respective functions and use cases, many of the applications that have been developed in the context of the Real Laboratory also provide REST API, allowing their core functions to be reused and orchestrated using the AI Framework OPACA, developed in the Go-KI project.

Many of these applications have been developed or extended by different student teams during the "Ambient Assisted Living" (AAL) course, offered by TU Berlin and supervised by members of ZEKI and the Go-KI project. The following sections describe each application in detail, including relevant research work and their specific implementation.

Multisensor

The multisensor platform is a cutting-edge environmental monitoring system developed to measure a diverse range of environmental parameters, including CO2, temperature, humidity, light intensity, air pressure, air quality, and noise levels. This platform is designed to deliver high-precision data acquisition in real time, enhancing environmental awareness and sustainability in smart workspaces. The integration of robust sensors and an efficient REST API ensures seamless operation and real-time data transmission across multiple rooms within the ZEKI environment.

Each multisensor is equipped with advanced components to ensure reliability and accuracy. The SCD30 CO2 sensor[2] provides high-precision carbon dioxide measurements, while the BME680 environmental sensor[3] monitors temperature, pressure, humidity, and gas levels. Ambient light intensity is measured using the VEML7700 sensor,[4] and noise levels are captured using a MAX4466 microphone with an amplifier.[5] The ESP32 microcontroller[6] serves as the central unit, coordinating data processing and wireless communication via its integrated WiFi and Bluetooth capabilities.

The system's scalability and modularity allow it to be deployed across diverse environments. At ZEKI, multisensors have been installed in most rooms to provide a comprehensive environmental overview. Data collected by these sensors is visualized in real time through integration with Home Assistant. The visualization overlays key metrics, such as temperature, CO2 levels, and humidity, on the ZEKI floor plan, enabling facility managers to address environmental issues proactively. Besides its built-in functions and the integration into Home Assistant, the Multisensor is also a key component used in several of the other Real Laboratory applications for getting environmental data.

[2] https://www.sensirion.com/en/environmental-sensors/carbon-dioxide-sensors/co2-sensor-scd30/
[3] https://www.bosch-sensortec.com/products/environmental-sensors/gas-sensors/bme680
[4] https://www.vishay.com/docs/84286/veml7700.pdf
[5] https://learn.adafruit.com/adafruit-agc-electret-microphone-amplifier-max4466
[6] https://www.espressif.com/en/products/socs/esp32

Recent advancements in multisensor technologies highlight the growing importance of environmental monitoring. Studies have emphasized the role of multisensor platforms in smart spaces to monitor air quality and energy efficiency (Astolfi et al., 2023), optimize indoor environmental conditions (McKenzie et al., 2009), and integrate sensor systems into urban environments (Terziyski et al., 2020). Moreover, AI-based approaches to analyze time-series environmental data (Hu et al., 2021, Liu et al., 2023) further enhance the value of multisensors by enabling predictive and proactive environmental management.

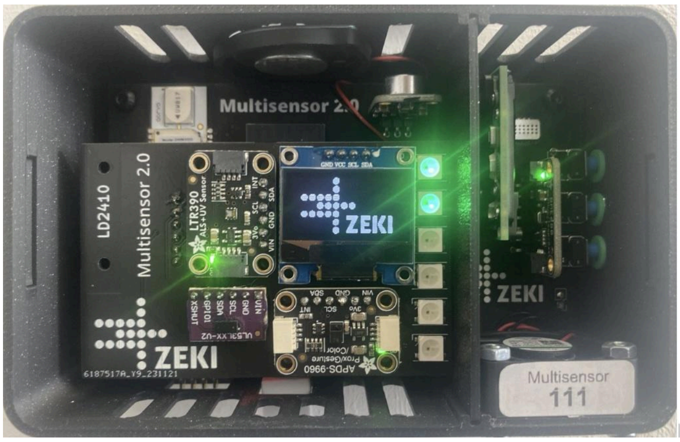

FIGURE: The multisensor platform: Key hardware components and their integration within the ZEKI environment.

Visualization of Environmental Data

The visualization module is a key feature of the multisensor platform, transforming raw data into actionable insights. Leveraging Home Assistant's capabilities, the system provides a user-friendly dashboard that overlays environmental metrics onto a spatial floor plan of ZEKI. This allows users to identify trends and anomalies in environmental parameters, such as CO2 levels, temperature, and humidity, in real time.

Each sensor's data is displayed in dedicated tabs for humidity, temperature, CO2, and pressure, ensuring clear and focused analysis. Additionally, individual sensor visualizations include current measurements, historical trends, and healthy ranges defined by guidelines of Bundesministerium für Arbeit und Soziales (BMAS). This ensures that all monitored parameters remain within safe limits for occupants, enhancing well-being and productivity.

FIGURE: Visualization dashboard: Overlay of environmental metrics on the ZEKI floor plan using Home Assistant.

Conversation Engine

The Conversation Engine is a pivotal component of the Real Laboratory, designed to enhance human-robot interaction (HRI) through adaptive and context-aware dialogues. It builds upon the integration of the humanoid robot “Pepper”[7] with advanced natural language processing (NLP) capabilities, powered by ChatGPT. We combined Pepper’s social robotics features with ChatGPT to provide an interactive and informative interface for visitors at the ZEKI (Smith et al., 2024). Their findings emphasized the importance of combining a conversational AI's linguistic proficiency with Pepper’s physical gestures and emotional intelligence to create a natural and engaging user experience (Jones et al., 2023).

[7] https://en.wikipedia.org/wiki/Pepper_(robot)

Implementation in the Real Laboratory

In the Real Laboratory, the Conversation Engine utilizes Pepper's onboard sensors for facial recognition and emotion detection. Integrated with ChatGPT via LangChain, the system supports complex task-oriented interactions. The Conversation Engine allows Pepper to assist visitors by answering queries about the Real Laboratory, ZEKI, activities, employees and discussing the ongoing projects. Real time web crawling ensures up-to-date responses about laboratory activities. Notable enhancements include memory features for conversational continuity and customized gestures to align with the context of the interaction. This innovation underscores the growing potential of combining physical robotics with conversational AI, enabling more accessible and relatable interactions between humans and intelligent systems. Future iterations aim to improve speech recognition precision and incorporate multilingual capabilities.

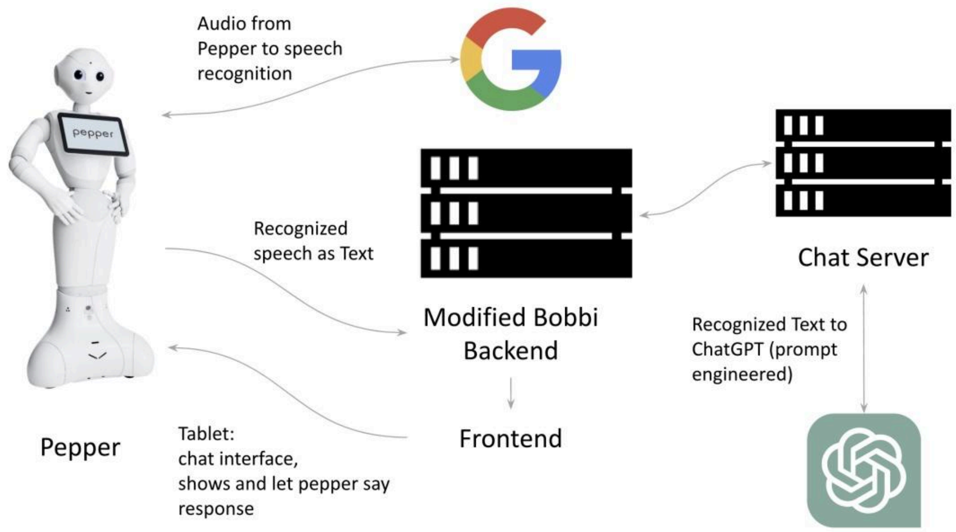

FIGURE: Conversation Engine's architecture and functionality illustration.

Smart Kitchen

The Smart Kitchen application in the Real Laboratory integrates machine learning (ML), object recognition, and web-based automation to redefine kitchen management and sustainability. By leveraging advancements in computer vision, such as the YOLOv8-World model for real-time object detection,[8] the system addresses four primary functions: waste management, recipe generation, inventory tracking, and shopping list creation.

The implementation employs ESP-32 cameras strategically installed in storage areas to capture images of groceries. These images are processed using the YOLOv8-Worldv2 model, which identifies items and updates an SQLite database with information such as expiration dates and categories. This database integrates with a web application, enabling users to track inventory, generate recipes dynamically, and create shopping lists. Recipes, generated by GPT-4, utilize available ingredients to minimize food waste. Additional features include automatic alerts for expiring items and the seamless integration of the LUNA voice assistant for hands-free operation.

[8] https://github.com/ultralytics/yolov8

The machine learning component of the Smart Kitchen leverages a fine-tuned version of YOLOv8-Worldv2. This model was specifically adapted to the kitchen environment by training it on the MVTec D2S dataset,[9] which contains densely segmented images of German groceries across various categories. Additionally, a custom dataset of 1,000 kitchen-specific images was manually annotated and incorporated to improve recognition accuracy under real-world conditions, such as cluttered arrangements and variable lighting. This fine-tuning process significantly improved the system's precision and recall for grocery detection tasks.

FIGURE: The Smart Kitchen setup showing ESP-32 camera placements and item recognition using YOLOv8-Worldv2.

The figure illustrates the Smart Kitchen setup, highlighting the placement of ESP-32 cameras within storage areas and the real-time object detection system. These cameras are strategically located to provide comprehensive coverage, ensuring accurate grocery recognition and inventory updates. The system’s ability to identify items under variable conditions is a core feature currently being optimized for enhanced performance.

The application also integrates a user-friendly web interface for efficient inventory management and meal planning. Users can scan items, monitor inventory, and receive real-time feedback on item availability or expiration alerts.

This Smart Kitchen's web interface is divided into functional panels. The left section demonstrates the grocery-scanning process with heatmaps for object detection, while the right section shows an organized inventory. Each item includes details such as quantity, expiration status, and category. The system ensures intuitive navigation and enhances user experience, simplifying inventory management and promoting sustainability.

LUNA, a multilingual voice assistant powered by Whisper and GPT-4, further enhances the Smart Kitchen's usability. Supporting 98 languages, LUNA facilitates hands-free interaction for a variety of tasks, such as opening shelves, locating items, or providing recommendations for storage. Equipped

[9] https://www.mvtec.com/company/research/datasets/mvtec-d2s

with a 360-degree microphone array[10] and a compact computing platform,[11] LUNA manages tasks like weather forecasts, recipe suggestions, and music playback. Its API allows seamless integration with new devices, ensuring scalability and adaptability in modern kitchens.

Challenges, such as maintaining model accuracy under variable lighting conditions, are being actively addressed. Future upgrades include optical character recognition (OCR) for text extraction from packaging and enhanced integration with smart home systems. These improvements aim to optimize waste management and simplify kitchen workflows, making the Smart Kitchen an essential part of sustainable living.

Interactive Wayfinding

Interactive Wayfinding is a critical application in the Real Laboratory, leveraging advanced technologies to enhance navigation within complex indoor environments. Inspired by the Ground Guiding Assistant System (GGAS), this application uses addressable LED strips installed on the floor to provide intuitive, visual navigation guidance. This innovative approach eliminates the need for users to carry additional hardware while ensuring accessibility and simplicity (Schulz et al., 2024).

Implementation in the Real Laboratory

The Interactive Wayfinding system in the Real Laboratory employs LED strips embedded within the floor pathways to guide users to specific locations. The system integrates touch and voice interactions via a variable user interface. Users can input their destination through a touchscreen map or voice commands, which are processed by an LLM to identify the requested location. Once a destination is selected, the system activates the corresponding LED strip segment with a designated color. Real-time animation on the strips dynamically guides the user along their route. The backend utilizes ESP32 controllers, which manage the LED animations and ensure seamless transitions between paths. These features make the system highly effective for multi-user navigation, allowing up to three users to be guided simultaneously with distinct color-coded paths.

This solution highlights the potential of combining physical guidance systems with advanced AI to create a robust, user-centric navigation system. Future expansions will explore incorporating motion tracking and adaptive LED animations to improve functionality in more dynamic settings.

[10] Seeed Studio ReSpeaker USB Microphone Array: https://wiki.seeedstudio.com/ReSpeaker_USB_Mic_Array
[11] Minis Forum GK41 Mini-PC: https://www.minisforum.com

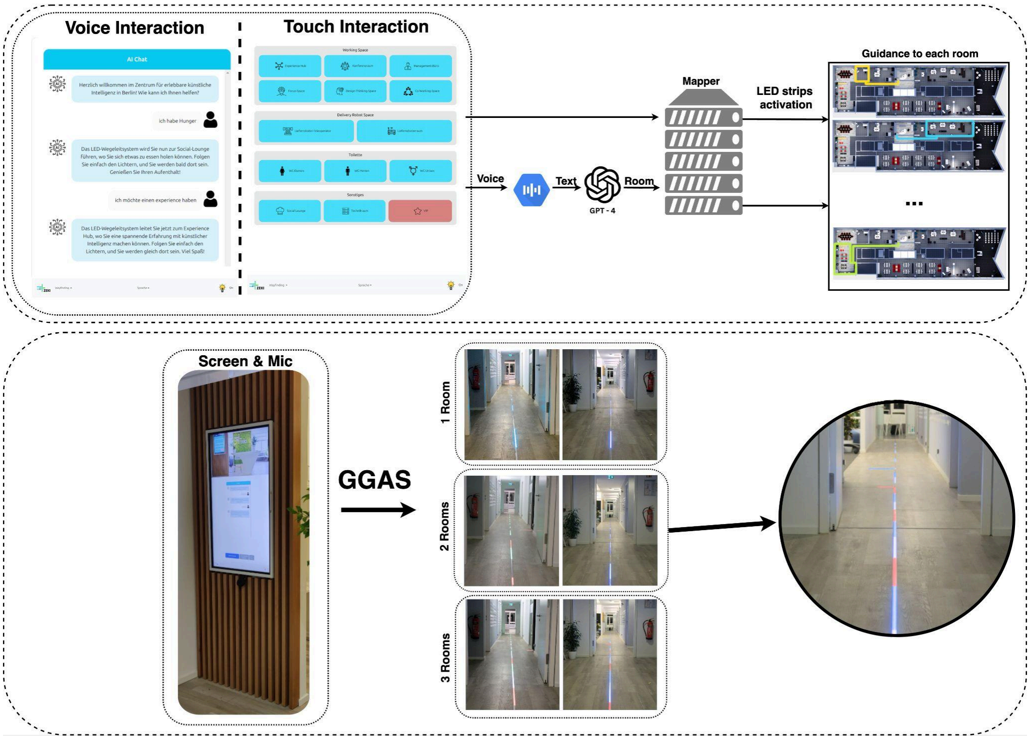

FIGURE: Voice and Touch interactive modes of GGAS and visualization of the wayfinding method

Invisible Interaction: Smart Ring for Gesture-based Control

The Invisible Interaction smart ring is a device designed to enable seamless, gesture-based interaction with smart environments. Leveraging capacitive proximity sensing and accelerometer data, this device integrates advanced gesture recognition algorithms to facilitate intuitive control without visible or intrusive interfaces. In the Real Laboratory setting, the Invisible Interaction ring has been deployed to control smart kitchen cabinets, demonstrating its potential for revolutionizing interactions in modern spaces.

The smart ring communicates via Bluetooth Low Energy (BLE) with a local processing unit, a Raspberry Pi, which serves as the gesture recognition hub. This Raspberry Pi processes real-time data from the ring's sensors to interpret specific gestures, such as swiping motions, and translates these into actionable commands. These commands are then sent to the ShelfAPI to execute tasks like opening or closing cabinets.

Thus, the Invisible Interaction system allows for control of kitchen cabinets with intuitive gestures, eliminating the need for physical interaction with handles, while being designed for easy expansion to support additional gestures and integrate with other smart home systems. All data are processed locally, ensuring user privacy by avoiding cloud-based data storage or external transmission.

In the current setup, four distinct gestures—swipe left, right, up, and down—are mapped to open or close specific cabinets. This implementation has proven effective in simplifying user interactions and enhancing accessibility, especially for individuals with mobility challenges. The following figure illustrates the smart ring in action, demonstrating its integration into the smart kitchen environment.

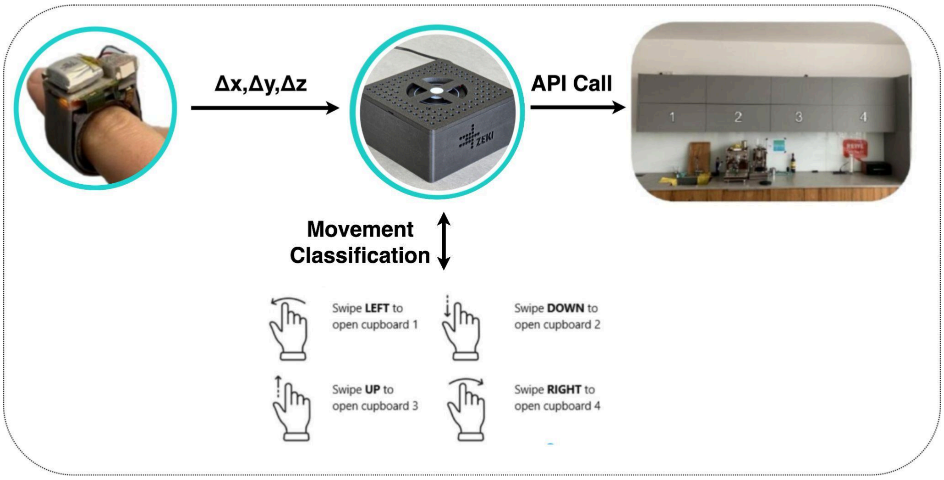

FIGURE: Invisible Interaction: Gesture-based control in the smart kitchen.

Future Developments

The Invisible Interaction project aims to further refine gesture recognition accuracy and expand its applicability. Upcoming enhancements include improved battery life, a more robust hardware design, and integration with additional IoT devices. Furthermore, dynamic gesture customization will allow users to define their own interactions, catering to personalized needs and preferences.

Smart Workplace: Ergonomic Desk and Chair and Furniture Box

The Smart Workplace initiative integrates the Ergonomic Desk and Smart Chair into a cohesive system designed to improve user health, productivity, and comfort. This innovative pairing demonstrates how ergonomic principles and IoT technologies can transform sedentary work environments into adaptive and health-conscious spaces (Gilson et al., 2011, Voordt et al., 2023). Alternatively, the Furniture Box is a small device that can be mounted to most office chairs to provide a subset of the functionalities of the Smart Desk and Chair.

Ergonomic Smart Desk

The Ergonomic Smart Desk exemplifies the integration of cutting-edge technology with practical workplace ergonomics, aimed at promoting healthier work habits and improving productivity. At the core of its functionality is a connection to the multi-sensor system that continuously monitors environmental parameters, including temperature, CO2 levels, humidity, and ambient light intensity. These real-time measurements are obtained directly from a centrally connected multi-sensor network, ensuring accurate data collection and reliability. The collected data is processed by a central application hosted on a server, making it accessible across multiple devices, including smartphones, computers, and a dedicated 9-inch tablet mounted to the desk. This centralization not only ensures real-time data synchronization but also supports seamless user interaction and configuration adjustments.

The interactive user interface, accessible through the mounted tablet, provides users with comprehensive environmental feedback. Key parameters are displayed in a color-coded format—green for optimal, orange for suboptimal, and red for poor conditions—helping users understand and respond to their workspace environment. The figure shows the user interface. This interface also includes additional functionalities, such as timers for work and breaks, ensuring adherence to recommended sitting and standing intervals, thus fostering dynamic working habits.

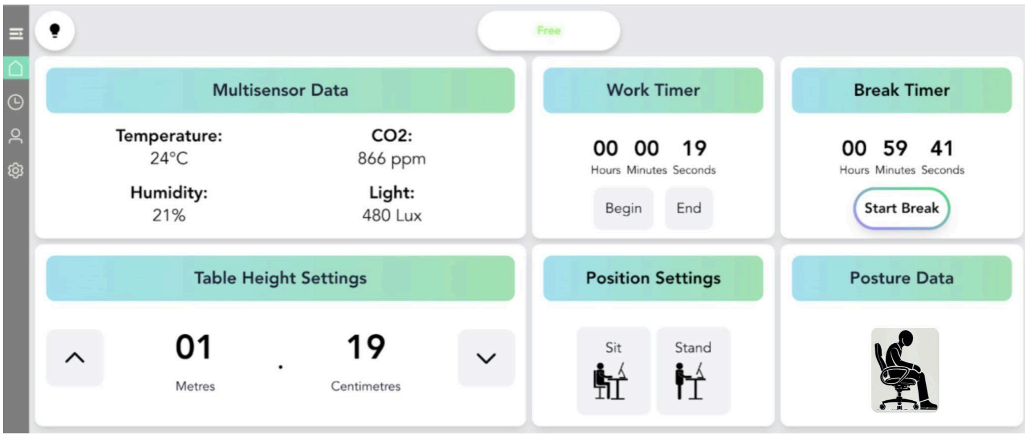

FIGURE: User Interface of the Ergonomic Smart Desk displaying environmental parameters and height-adjustment settings.

The desk integrates a height-adjustable system powered by a reverse-engineered ELIOT desk motor. By decoding the proprietary protocol, custom commands for raising and lowering the desk have been implemented, enabling precise control. Users can save personalized height preferences, which are automatically applied upon logging into the desk system via the application. This feature ensures consistency and ease of use in shared office spaces, catering to individual ergonomic needs. Additionally, LED indicators embedded in the desk can be used for e.g. visual cues for its reservation status or current occupant, making it ideal for shared or co-working environments. These indicators allow quick identification of availability, further streamlining desk utilization.

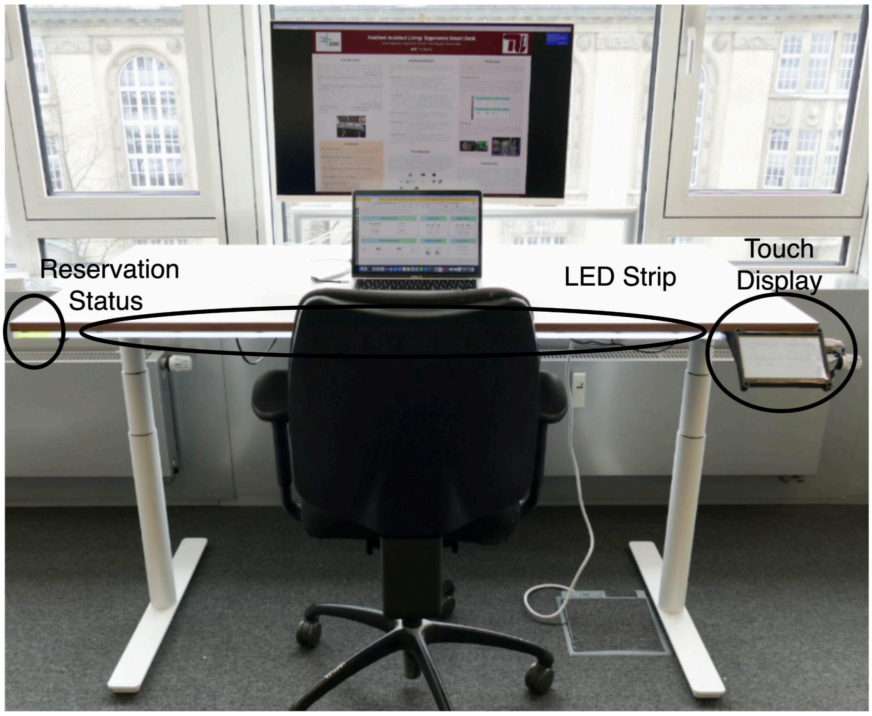

FIGURE: Overview of the Ergonomic Smart Desk setup with the mounted tablet and interactive components.

Similar efforts have explored smart desk systems with embedded sensors and ergonomic features. For instance, Aryal et al. (2019) discussed a multi-sensor desk for monitoring workspace parameters, while Hedayati et al. (2019) and Johnson et al. (2020) investigated adjustable desks for promoting workplace health. Further, technologies for real-time environmental feedback in shared spaces have been elaborated by Kim et al. (2019) and Chen et al. (2021), emphasizing the growing importance of integrated smart furniture in modern offices.

Smart Chair

The Smart Chair complements the desk by focusing on posture monitoring and correction. It utilizes MD30-60 pressure sensors connected to an ESP32 microcontroller to evaluate the user's posture in real time (Kundaliya et al., 2023). Eleven predefined postures, including "unoccupied," are identified and transmitted as JSON data to the Ergonomic Desk or other connected devices. This capability fosters healthy seating habits by offering immediate feedback and guidance to users. Additionally, the chair's design supports potential enhancements such as long-term posture profiling, which could provide valuable insights for addressing chronic seating-related health issues (Voordt et al., 2023).

The integration of the Ergonomic Desk and Smart Chair creates a synchronized system that holistically monitors and improves the user's workspace experience. Data collected by the chair complements the desk's feedback mechanisms, providing a unified overview of posture and environmental conditions. This synergy exemplifies the potential of IoT-enabled furniture to enhance health, productivity, and user satisfaction in modern work environments (Aryal et al., 2019, Gilson et al., 2011).

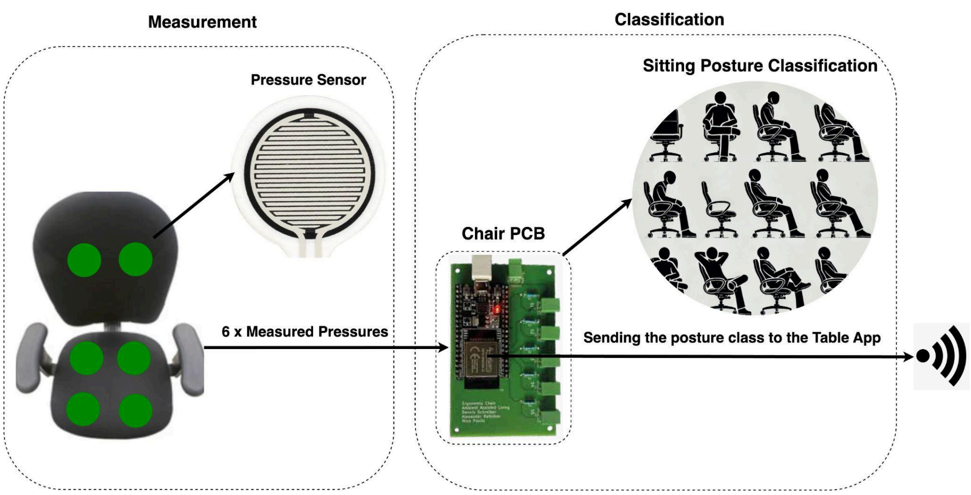

FIGURE: Smart Chair equipped with pressure sensors for real-time posture monitoring.

Furniture Box

The Furniture Box is an innovative application in the Real Laboratory, designed to enhance user ergonomics and environmental awareness in smart workspaces. This compact device, which can be mounted on the arm of a chair, integrates proximity sensors and an eInk LCD display to provide real-time feedback on sitting duration and environmental conditions. Inspired by modern ergonomic systems and IoT-based smart environments (Kim et al., 2018, Gupta et al., 2023), the Furniture Box acts as a personal workspace assistant.

The Furniture Box utilizes a proximity sensor to monitor user presence and calculate the total time spent seated. This data is displayed on an energy-efficient eInk screen, offering users clear and non-intrusive feedback. Additionally, the device directly retrieves air quality data, such as CO2 levels and particulate matter (PM2.5), via WiFi from environmental sensors integrated into the workspace network. The collected data ensures users are aware of their workspace's air quality, promoting healthier and more productive habits. Alerts are displayed on the LCD screen if air quality falls below predefined thresholds.

This system operates autonomously, relying on real-time sensor communication without requiring intermediary devices. Its portable design allows it to adapt to various workspace configurations, providing a seamless user experience. Future improvements may include adaptive feedback mechanisms that integrate ergonomic suggestions based on user posture and extended usage trends.

Smart Planting: Sensor-based Plant Care in the Office

The Smart Planting system is an innovative solution designed to automate and enhance indoor plant care using advanced sensor technologies and seamless integration with smart home systems. This system consists of a custom-designed PCB incorporating essential sensors for soil moisture, light intensity, temperature, humidity, and atmospheric pressure. It provides continuous monitoring and intuitive feedback to ensure optimal plant health. The integration with Home Assistant enables real-time notifications and a user-friendly web interface for managing plant care.

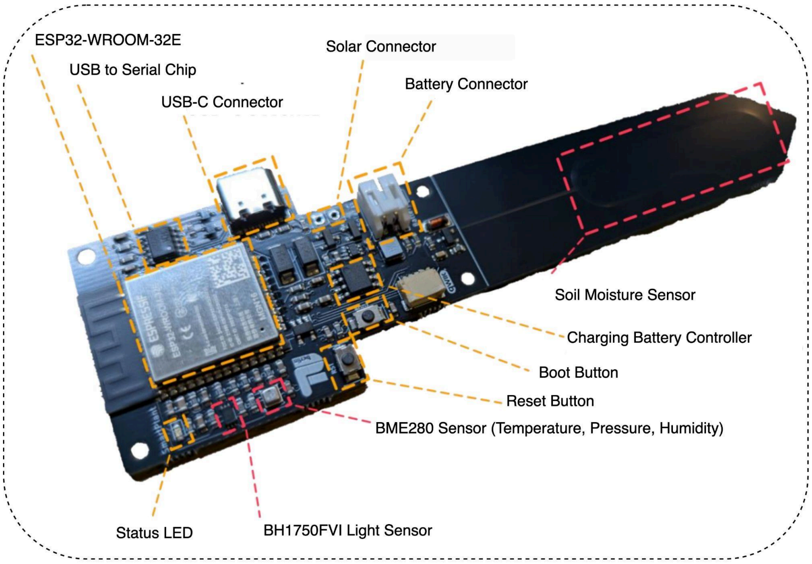

FIGURE: Overview of the Smart Planting sensor with its components labeled.

The core hardware comprises: An ESP32-WROOM-32E[12] microcontroller with built-in WiFi for real-time data transmission; a BME280 Sensor[13] for measuring temperature, humidity, and pressure; a BH1750FVI Light Sensor[14] to assess current light conditions; a Soil Moisture Sensor[15] to determine

[12] https://www.espressif.com/en/products/socs/esp32
[13] https://www.bosch-sensortec.com/products/environmental-sensors/humidity-sensors-bme280
[14] https://www.mouser.com/datasheet/2/348/bh1750fvi-e-186247.pdf
[15] https://www.dfrobot.com/product-1130.html

watering needs; a Solar and Battery Charging Circuit (CN3063)[16] for being able to operate on solar power or USB charging; and a USB-C Connector and Status LED[17] to facilitate connectivity and indicate operational status.

This system integrates seamlessly into the Home Assistant dashboard, displaying real-time sensor data in an easily interpretable format. Users can customize thresholds for each parameter and receive alerts on their devices when values deviate from the optimal range. The compact PCB design, with dimensions suitable for most potted plants, ensures unobtrusive placement while maximizing functionality.

The Smart Planting system builds on existing research while addressing gaps in commercial solutions. For example, unlike Bluetooth-only devices like PlantLink and Parrot Flower Power,[18] this system operates over WiFi and integrates directly with smart home systems, enabling remote monitoring and automation of plant care tasks.[19] Moreover, it improves upon agricultural IoT systems by offering a scalable, low-maintenance design tailored for indoor use. By utilizing solar power and a rechargeable battery, it eliminates the need for frequent maintenance, making it a sustainable option for long-term use (Khaled et al., 2022).

In summary, the Smart Planting system represents a step forward in indoor plant care, combining cost-effective design, robust sensor integration, and user-centric software to provide a reliable and scalable solution for homes and offices.

Transferability

The applications developed for the Real Laboratory demonstrate significant potential for transferability to new environments. This section elaborates on how each application can be adapted, the specific steps required for their deployment, and the extent of customization or reconfiguration needed to ensure seamless operation in a different space.

Multisensor

The transferability of the multisensor platform to a new space involves replicating its environmental monitoring capabilities while addressing the unique physical and operational requirements of the target location. The hardware, including sensors such as the SCD30, BME680, and VEML7700, can be redeployed with minimal changes. However, the following customizations are necessary:

- Calibration: Sensor recalibration is required to align with the environmental conditions of the new space, such as altitude (for air pressure) or baseline CO2 levels.
- Visualization Updates: The Home Assistant dashboard must be updated with a new floor plan reflecting the layout of the target environment. This ensures accurate spatial representation of sensor data.

[16] https://datasheet.lcsc.com/lcsc/2201071335_Shenzhen-Chip-On-Microelectronics-Tech-CN3063_C28637.pdf
[17] https://www.sparkfun.com/products/15794
[18] https://www.parrot.com/assets/s3fs-public/2021-09/flower-power_user-guide_uk.pdf
[19] https://www.cnet.com/reviews/oso-technologies-plantlink-review

Conversation Engine

The Conversation Engine's transferability hinges on adapting Pepper's interactions to the context of the new space. While the hardware and core AI capabilities remain the same, significant adjustments are required:

- Knowledge Base Update: Pepper's integrated ChatGPT system must be updated with contextual information about the new environment, including its layout, personnel, and services.
- Localization: Multilingual capabilities may need to be expanded based on the predominant language of the new space. Speech recognition and synthesis modules must be fine-tuned for regional accents or dialects. Similarly, Pepper's gestures and behaviors should align with the cultural norms and expectations of the new audience.
- Additionally, real-time web crawling configurations should be updated to fetch relevant information specific to the new space.

Smart Kitchen

Transferring the Smart Kitchen application involves ensuring compatibility with the new space's layout, appliances, and network infrastructure. The following steps are critical:

- Camera Placement: The ESP-32 cameras must be strategically installed in storage areas of the new kitchen to ensure complete coverage and optimal performance of the object recognition system.
- Dataset Expansion: If the new space includes groceries or items not present in the initial dataset, additional images and annotations must be added to retrain the YOLOv8-World model for accurate detection.
- Integration with Local Appliances: The Smart Kitchen must be connected to existing appliances through IoT protocols. Compatibility testing and configuration adjustments may be needed for devices like refrigerators, ovens, or smart shelves.
- UI Adaptation: The web application interface should be customized to reflect the inventory and workflows specific to the new kitchen setup.
- The core object detection and inventory tracking functionalities remain robust but may require minor tuning for new environmental conditions.

Interactive Wayfinding

To deploy the Interactive Wayfinding system in a new indoor environment, modifications are necessary to account for differences in layout and user requirements:

- Pathway Mapping: The LED strip pathways must be redesigned to fit the layout of the new space. This involves physical installation and updating the backend to reflect the revised pathways.
- Backend Configuration: ESP32 controllers managing LED animations must be reprogrammed to accommodate the updated routes and colors assigned to specific locations.
- User Interface Update: The touchscreen map and voice command interface need to include the new space's destination points and contextual information.

- The current design only works for a single floor; if wayfinding across multiple floors should be supported, it might be necessary to install and connect multiple instances.

Invisible Interaction (Smart Ring)

The Smart Ring's transferability primarily involves adapting its gesture recognition system to the devices in the new space:

- Gesture Mapping: New gestures may need to be defined to control different devices in the new environment, requiring updates to the gesture recognition algorithm.
- Device Integration: The ring's Bluetooth communication must be configured to interface with the IoT devices in the target space, such as smart doors, lights, or appliances.
- User Training: Users in the new space may require training to familiarize themselves with the gesture-based control system.

Smart Workplace

The Ergonomic Desk and Smart Chair system can be transferred with minimal hardware changes, but reconfigurations are necessary:

- User Profiles: Personalized height and posture preferences stored in the application must be reset or migrated for new users.
- Visualization Update: The tablet interface should be customized to reflect the environmental parameters and ergonomic guidelines specific to the new space.

Furniture Box

The Furniture Box's portability makes it inherently transferable, but some reconfiguration is required:

- Air Quality Data Source: In a new space, the Furniture Box must connect to the local multisensor network for real-time environmental data. API endpoints for data retrieval may need to be updated.
- Threshold Adjustment: Alert thresholds for sitting duration and air quality parameters may need to be adjusted based on the guidelines applicable to the new environment.

Smart Planting System

The Smart Planting system’s transferability involves the following adjustments:

- Sensor Calibration: Soil moisture, light, and environmental sensors must be recalibrated for the conditions of the new location.
- Dashboard Integration: The Home Assistant dashboard must be updated to reflect the new plant species and their specific care requirements.
- Power Management: Solar charging and battery configurations should be optimized for the light conditions of the new space.

AI Framework Integration

Besides the above application, an AI framework has been developed in the course of the Go-KI project, combining aspects of multi-agent systems with microservices and containerization. This allows for the creation of heterogeneous and distributed systems using a common API. In addition to the framework, different development and runtime tools were created, providing for an easier service development and more intuitive interaction. In particular, those include a BPMN editor and interpreter specifically adapted to the framework's API, and an LLM-based assistant, allowing developers and users alike to use natural language queries to interact with the running system.

In combination, these tools bridge the technical gap by integrating process modeling, allowing non-technical domain experts to create service orchestrations without needing to write code, and they lower the barrier for all users of the system.

The OPACA Framework

The "OPACA" framework (Open, Language- and Platform-Independent API for Containerized Agents) combines concepts of multi-agent systems with modern container and microservice technologies (cf. Acar et al., 2024). The framework defines two primary components: Agent Containers (AC), providing Agents that can offer different Actions, react to external events or exhibit proactive behavior, and Runtime Platforms (RP), managing one or more Agent Containers, and providing basic functionalities, e.g., token-based authentication. The Agent Containers are running in Docker or Kubernetes, while the Runtime Platform connects them with each other and possibly with other Runtime Platforms.

The ACs and RPs communicate with each other and with the outside world or external tools via a standardized REST API, providing routes to get currently running agents and their actions, send messages to agents, or invoke actions. The reference implementation[20] uses Java with Spring Boot for the Runtime Platform and Kotlin with JIAC VI (Rakow, 2019) for the Agent Containers; however, individual ACs can be implemented in any programming language as long as they follow the API. While the main function of the ACs is to provide Actions that can be called as webservices, each AC can also expose additional ports for, e.g., hosting its own, possibly complex, web UI.

OPACA BPMN Editor

A popular way to orchestrate existing services to more complex applications is using BPMN processes (OMG, 2011, Küster et al., 2016), combining an intuitive notation with rich execution semantics. It provides an excellent compromise between ease of use, expressiveness, and self-documentation, with many applications in industry. With OPACA's uniform API, it is particularly easy to search and combine actions provided by the agents running on the platform and its containers. Thus, in the course of the Go-KI project, a new BPMN editor and interpreter have been developed,[21] based on the *BPMN-js* framework.[22]

[20] OPACA Framework Source Code: https://github.com/GT-ARC/opaca-core
[21] OPACA BPMN Editor Source Code: https://github.com/gt-arc/opaca-bpmn
[22] BPMN.io: https://github.com/bpmn-io/bpmn-js

While BPMN-js provides an excellent web-based BPMN editor, by itself it does not allow for creating executable processes. For this, additional elements for e.g. variables and assignments had to be defined using BPMN extensionElements, as well as a service element for describing OPACA actions, both in the editor's model and user interface. Further, the editor was extended with a Services list, which allows connecting to a running OPACA RP and importing all its actions as services into the BPMN diagram. This way, the OPACA actions can be bound to Service Tasks and be combined to new, complex executable applications.

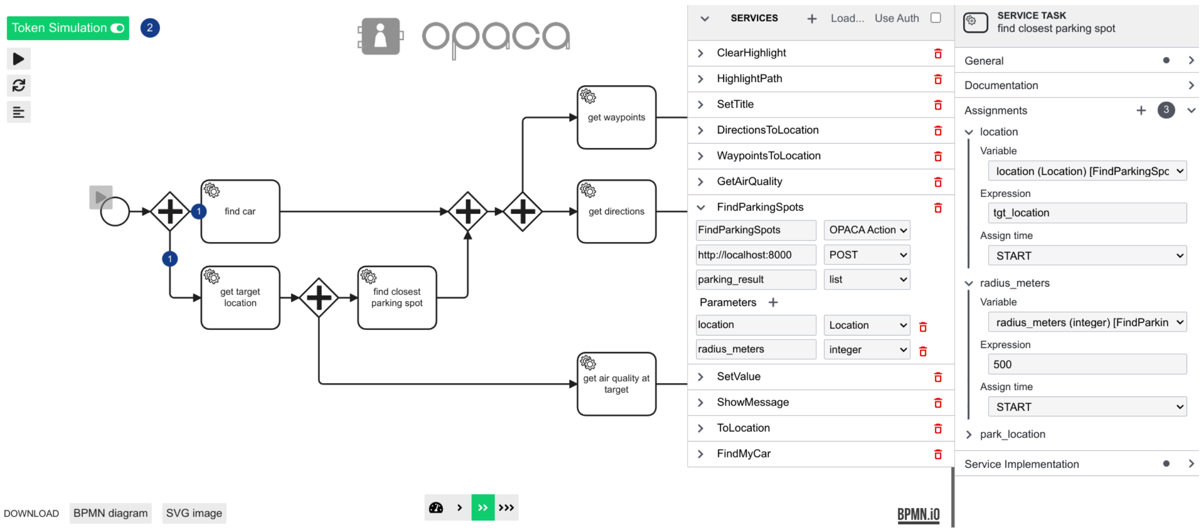

FIGURE: OPACA BPMN Editor, showing main editing area (with active token simulation), list of connected services, and selected element's properties.

The process interpreter uses the BPMN-js' *Token Simulation*, which is integrated into the editor itself (at the same time using it as a monitoring tool). Like the editor, it had to be extended to support variables, assignments, and service calls, in order to automatically evaluate conditions and invoke OPACA actions. The interpreter can be used from within the BPMN editor, or while running in a "virtual web browser" within an OPACA Agent Container, allowing it to load and execute several BPMN diagrams at once in "headless" mode.

OPACA LLM Support

In addition, a specifically prompted Large Language Model (LLM) can be used for the interaction with an OPACA platform. Having knowledge about the agents running on an OPACA platform and their actions, and being able to call them at the request of the user, it provides a more ad-hoc, intuitive, and barrier-free interaction than the BPMN editor.

The OPACA LLM Integration consists of two parts: A frontend, implemented in Javascript using Vue.js, providing a web UI mainly consisting of a large "chat" window, but also including controls for configuring the OPACA platform and LLM to use, as well as some multimedia support (speech and images), and a backend, implemented in Python, taking care of the message history per session, using FastAPI to accept requests from the frontend, and implementing different methods for interacting with the actual LLM and the connected OPACA platform.[23]

[23] OPACA LLM Source Code: https://github.com/gt-arc/opaca-llm-ui

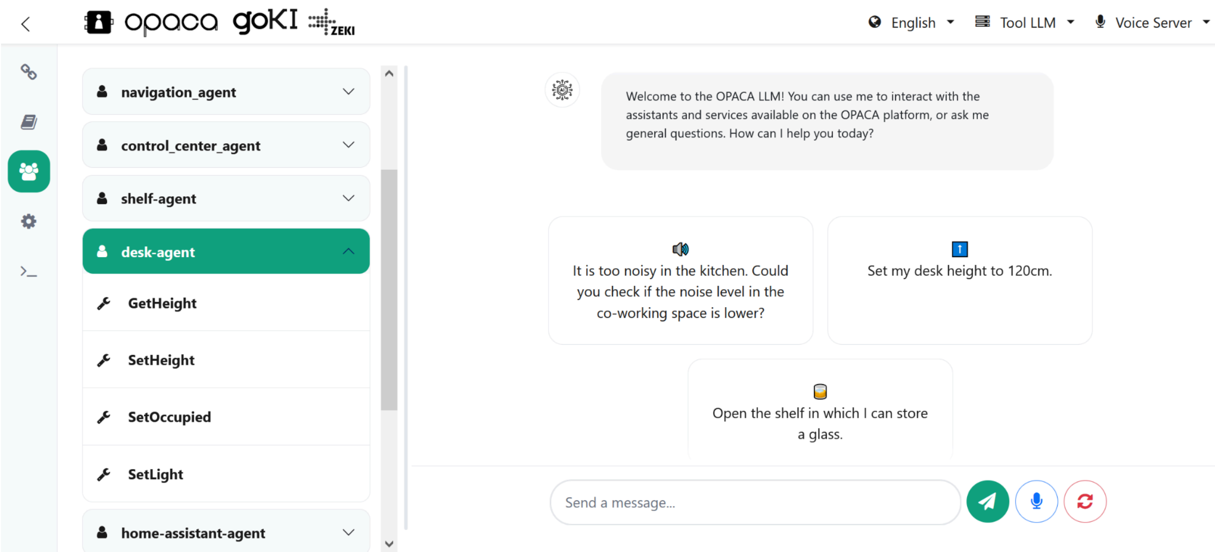

FIGURE: OPACA LLM UI, showing list of available agents and services, and main chat area.

For the interaction with the actual LLM, several methods have been implemented:

- A simple approach with a single LLM agent,[24] passing the available actions directly in the prompt and searching the LLM's response for action invocations;
- A more complex approach based on REST-GPT (Song et al., 2023), including four different LLM agents for planning, selecting concrete actions, calling them and evaluating the results;
- An approach using two LLM agents for generating and evaluating action invocations, using the "tools" feature found in most, but not all, more recent LLMs;
- An approach where several groups of LLM agents are dynamically created for each agent in the OPACA framework, each responsible for that agent's actions, as well as other agents for orchestrating which of those agents to use and for evaluating the results.

In addition, each of the approaches can be configured to use different LLMs, such as GPT, LLAMA, Mistral, and others, as well as set common parameters such as the LLM's "temperature". Further, all methods can go into additional iterations before returning a result to the user in case multiple actions have to be called, and they are also able to help the user with general LLM capabilities, such as some "common knowledge", text summarization, etc. – capabilities that are also used by the LLM to "guess" appropriate parameters for actions, if not explicitly specified, or for evaluating and summarizing the results before presenting them to the user.

While the OPACA LLM is more restricted in the kinds of services it can execute, compared with other more generic approaches, the uniform service descriptions of OPACA actions also makes the LLM integration simpler and more robust: In our tests the LLM was able to make sense of and correctly call and combine all services it was presented with without special training, fine-tuning, or extensive examples, making it well suited for the highly dynamic service discovery mechanisms provided by the OPACA framework.

[24] Note that in this section, the term "agent" can refer to either an agent in the OPACA framework, or an LLM agent, i.e. a part of the LLM pipeline with its own purpose, prompt, and message history.

Integration with Real Laboratory Applications

For the integration with the ZEKI Real Laboratory, several of the applications' functions have been made available as OPACA Agent Containers. For many applications that provide a REST API of their own, e.g. the Multisensor and Smart Kitchen, this has been done by creating a thin OPACA "proxy" agent, providing actions that internally just delegate to the actual Real Laboratory applications, with some additional logic for making them easier to use (e.g. an internal mapping of rooms to their respective IDs). Other services have been developed directly in OPACA, providing additional functionalities that can be used along with the other Real Laboratory applications.

The following applications related to Go-KI and the Real Laboratory are currently available in OPACA:

- The above-mentioned proxy to several of the Real Laboratory applications, including data from the Multisensors (temperature, humidity, CO2 and noise in all rooms at ZEKI), control over the shelves in the kitchen, controlling the Smart Desk, and the Wayfinding system.
- A basic Personal Information Management service, interfacing with the Exchange API to provide access to services such as e-mails, calendars, and contacts.
- Information services for e.g. stock prices, weather information, and routing/navigation, making use of different external APIs (all free), as well as summarization services that can be used to search Google, Wikipedia, or open a specific website and use another internal LLM to summarize the contents.
- Visualisation services that can be used for showing textual or tabular data on a simple web UI, and for generating different types of charts and diagrams for inputted data.
- A proxy for several of the functions of the BeIntelli API[25] that can be used to e.g. find the location of different vehicles in the BeIntelli fleet, read values from roadside sensors or find free parking spots in the vicinity of ZEKI.
- Another proxy-agent forwarding the given requests to two other LLMs specialized on answering questions around the Berlin administration as well as summarizing and visualizing historic data on e.g. traffic incidents or the past Covid pandemic.

A test deployment of the OPACA system, the tools and above applications has been installed at ZEKI (currently only accessible from within ZEKI's intranet), where it has been used, without major interruptions over a course of several months, for interacting with the environment by both, developers and visitors or ZEKI. In this setting, we have made the following observations:

- Developers who were not familiar with the framework or tools were able to quickly and correctly implement new applications and to integrate them into the deployed system.
- The tools, esp. the LLM UI, were easy to use by co-workers and visitors without any training.
- The BPMN Editor required some amount of understanding of the notation (and development, for executable processes), but was otherwise easy and intuitive to use.
- The different LLMs were able to "understand" and work with the different services in the test deployment without requiring any additional training, fine-tuning or examples.
- The BPMN Editor has been used to create working and useful processes for the BeIntelli project that otherwise would have needed to be implemented by other means.

[25] BeIntelli API: https://be-intelli.com/apis-und-sdks/

While this does not yet constitute a full and systematic evaluation, it strongly indicates the range of applications that can be created with the AI framework and tools, and the ease with which they can be used even by untrained personnel.

Transferability

To apply the OPACA framework and tools in a different setting, clearly several of the application containers would have to be re-written – or in some cases just re-configured – to accommodate the new setting. However, given the language independence of OPACA and reliance on open and established standards, new Agent Containers can easily be created by either proxying an existing API (and enriching it with the extensive self-description of OPACA's agents and actions), or providing the service directly in an OPACA container, in Java, Python, JavaScript, or any other language.

The framework and tools themselves can quickly be set up on any system using Docker-Compose: All components include a Docker configuration that can be used to create a self-contained Docker container of that component and to combine them all in a Docker-Compose file. The LLM-Support can use different LLMs: Depending on the setting, confidentiality, and installed compute resources, it can use either OpenAI's GPT family, or a wide range of self-hosted open-source models such as LLAMA, Mistral, and others, using vLLM[26] or LiteLLM[27] as a proxy.

[26] vLLM: https://github.com/vllm-project/vllm
[27] LiteLLM: https://www.litellm.ai

Summary

In this report, we described the ZEKI work environment, the integrated Real Laboratory and the applications that have been developed in that context, as well as the AI Framework “OPACA” and its related tools.

- ZEKI is a highly dynamic and versatile workspace adjacent to TU Berlin and home to the Real Laboratory developed in the Go-KI project; it includes several innovative concepts and presents a space where AI researchers can develop new applications and experience those applications first hand in their daily work.
- The applications, most of which include a hardware and a software component, range from combined sensory units to smart furniture automatically adapting to the users’ needs; they were developed in part by students in the Ambient Assisted Living (AAL) course held at TU Berlin, and in part directly in the Go-KI project.
- Many of the applications have been developed with, or integrated into, the AI Framework OPACA, where they can be easily interacted with or orchestrated to more complex applications.

These components make ZEKI and the Real Laboratory an example for how a work-environment of the future may look like that harnesses digitalization and AI technologies to improve productivity and comfort. The tools and applications presented in this report may serve as a blue-print how similar environments can be set up, too: While most of the Real Laboratory applications have been developed with ZEKI in mind, this report shows how they can be adapted to be used in a different environment. The OPACA-Framework and tools have been designed from the outset to be as open and versatile as possible, further helping with the development and integration of similar Real Laboratory applications and assets in any given work environment.

We acknowledge that every work environment is different: ZEKI is an open-floor concept for collaborative and creative office work, while other work spaces may focus on individual offices, but most of the applications developed in the project would still be applicable. Other areas of work, such as care (e.g. hospitals), retail (e.g. department stores) or services (e.g. banks, lawyers), could also profit from some of the applications, while others may not be suited. On the other hand, workplaces focusing on more manual labour (workshops, factories, farms) will require a different set of applications.

References

- Y. Pan, "Living Labs Contributions to Smart Cities from a Quadruple-Helix Perspective," Journal of Science Communication, 2023. DOI: 10.22323/2.22030202.
- T. Sasaki and H. Hashimoto, "Intelligent Space as a Platform for Human Observation," Human-Robot Interaction Conference, 2007. DOI: 10.5772/5199.
- G. Diraco, A. Leone, and P. Siciliano, "AI-Based Early Change Detection in Smart Living Environments," Sensors, 2019. DOI: 10.3390/S19163549.
- F. Rivera-Illingworth, V. Callaghan, and H. Hagras, "Automated Discovery of Human Activities Inside Pervasive Living Spaces," International Symposium on Pervasive Computing and Applications, 2006. DOI: 10.1109/SPCA.2006.297520.
- H.-D. Bui and N. Y. Chong, "An Integrated Approach to Human-Robot-Smart Environment Interaction Interface for Ambient Assisted Living," Advanced Robotics and Its Social Impacts, 2018. DOI: 10.1109/ARSO.2018.8625821.
- D. Rodríguez and A. Natalia, "Semantic and Fuzzy Modelling for Human Behaviour Recognition in Smart Spaces," Dissertation, 2015.
- M. Vasileiadis, D. Giakoumis, K. Votis, and D. Tzovaras, "A Living Lab Infrastructure for Investigating Activity Monitoring Needs in Service Robot Applications," Pervasive Technologies Related to Assistive Environments, 2016. DOI: 10.1145/2910674.2935830.
- A. Braun, S. Rus, and M. Majewski, "Invisible Human Sensing in Smart Living Environments Using Capacitive Sensors," Springer Series on Ambient Intelligence, 2017. DOI: 10.1007/978-3-319-52322-4-3.
- T. Sasaki and H. Hashimoto, "Design and Implementation of Intelligent Space: A Component Based Approach," Book Chapter, 2010. DOI: 10.5772/8921.
- P. Malik, "The Integration of Natural Language Processing (NLP) in Human-Robot Interaction (HRI)," Indian Scientific Journal of Research in Engineering and Management, 2024. DOI: 10.55041/ijsrem35803.
- S. Rezwan, W. Ahmed, M. A. Mahia, and M. R. Islam, "IoT Based Smart Inventory Management System for Kitchen," International Conference Advances Computing, Communication and Automation, 2018. DOI: 10.1109/ICACCAF.2018.8776761.
- Z. Qiu, M. Ashour, X. Zhou, and S. Kalantari, "NavMarkAR: A Landmark-based Augmented Reality (AR) Wayfinding System," arXiv.org, 2023. DOI: 10.48550/arxiv.2311.12220.
- B. Prueksanusak, P. Rujivipatand, and K. Wongpatikaseree, "An Ergonomic Chair with Internet of Thing Technology using SVM," Proceedings Article, 2019. DOI: 10.1109/TIMES-ICON47539.2019.9024488.

- T. B. Brown, B. Mann, N. Ryder, et al., "Language Models are Few-Shot Learners," arXiv preprint arXiv:2005.14165, 2020.
- OpenAI, "Rate limits and API integration," Retrieved from: https://platform.openai.com/docs/guides/rate-limits, 2023.
- D. Hasiwar, A. Gruber, C. Dragschitz, and I. Ivkic, "Towards a Cloud-based Smart Office Solution for Shared Workplace Individualization," arXiv.org, 2024. DOI: 10.48550/arxiv.2403.18883.
- A. Joshi, S. T., T. M., J. Singh, and D. P., "IOT Based Smart Plant Monitoring System," International Journal for Research in Applied Science and Engineering Technology, 2024. DOI: 10.22214/ijraset.2024.58953.
- T. Cheng, Y. Liu, and X. Zhang, "YOLO-World: Real-Time Open-Vocabulary Object Detection," arXiv preprint arXiv:2401.17270, 2024.
- P. Follmann, R. König, P. Haubner, and M. Ulrich, "MVTec D2S: Densely Segmented Supermarket Dataset," arXiv preprint arXiv:1804.08292, 2018.
- V. Guimarães, H. Tan, and A. Costa, "A Review of Recent Advances and Challenges in Grocery Label Detection and Recognition," Applied Sciences, vol. 13, no. 5, 2023. DOI: 10.3390/app13052871.
- T. Schulz, P. Babakhani, M. Heine, F. Harms, M. G. Augusto, and S. Albayrak, "GGAS: Ground Guiding Assistant System for Interactive Indoor Wayfinding," in Proceedings of the 14th International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2024.
- Y. Xu, H. Wang, and J. Tan, "Augmented Reality for Indoor Wayfinding: A Systematic Review of Techniques and Applications," in IEEE Transactions on Visualization and Computer Graphics, vol. 30, no. 2, 2024, pp. 1234-1248.
- L. Zhao, C. Guo, and J. Liu, "NaviLight: Indoor Localization and Navigation Framework Using Visible Light," in IEEE Sensors Journal, vol. 17, no. 8, 2017, pp. 2508-2515.
- J. Kim and H. Park, "IoT-based Smart Chair for Personalized Ergonomic Monitoring," in Journal of Industrial Ergonomics, vol. 32, no. 4, 2018, pp. 367-375.
- A. Gupta, M. Ramesh, and K. Chandrasekaran, "Sensor-Driven Air Quality Monitoring Systems for Smart Environments," in IEEE Sensors Journal, vol. 23, no. 2, 2023, pp. 234-245.
- N. D. Gilson, M. Pratt, B. E. Hurst, L. E. Straker, and N. D. Ridgers, "Occupational sitting time: employees' perceptions of health risks and intervention strategies," Health Promotion Journal of Australia, vol. 22, no. 1, pp. 38–43, 2011.
- B. Kundaliya, M. Sharma, and P. Kumar, "An IoT and Cloud Enabled Smart Chair for Detection and Notification of Wrong Seating Posture," Preprint. Research Square, accessed Nov. 1, 2023.

- T. V. D. Voordt and P. A. Jensen, "The impact of healthy workplaces on employee satisfaction, productivity and costs," Journal of Corporate Real Estate, vol. 25, no. 1, pp. 29–49, 2023.
- A. Aryal, B. Becerik-Gerber, F. Anselmo, S. C. Roll, and G. M. Lucas, "Smart Desks to Promote Comfort, Health, and Productivity in Offices: A Vision for Future Workplaces," Frontiers in Built Environment, vol. 5, p. 76, 2019.
- K. Obaideen et al., ``An overview of smart irrigation systems using IoT,'' Energy Nexus, vol. 7, pp. 100124, 2022. [Online]. Available: 10.1016/j.nexus.2022.100124
- A. Gebhart, Oso Technologies PlantLink review: This simple sensor will give your plant a voice, 2014.
- Smith, J., Brown, A., and Lee, K. "Improving Autonomy and Natural Interaction of Pepper Robot via Large Language Models." Research Square, 2024. DOI: 10.21203/rs.3.rs-3997840
- Jones, R., Patel, S., and Kim, H. "An Exploration of the Pepper Robot's Capabilities: Unveiling Its Potential." Applied Sciences, vol. 14, no. 1, pp. 1010-1020, 2023.
- Cheng, Wei, Zhao, Ling, Sun, Xuan, "YOLO-World: Enhancing Real-Time Object Detection for Everyday Applications," IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 4, pp. 1234--1248, 2024.
- Smith, John, Taylor, Emma, "LUNA: A Multilingual Voice Assistant for Smart Home Applications," Journal of Artificial Intelligence and Smart Systems, vol. 18, no. 2, pp. 99--110, 2023.
- Khaled Obaideen, Bashria A.A. Yousef, Maryam Nooman Al Mallahi, Yong Chai Tan, Montaser Mahmoud, Hadi Jaber, Mohamad Ramadan: An overview of smart irrigation systems using IoT, Energy Nexus, Volume 7, 2022, 100124, ISSN 2772-4271, 10.1016/j.nexus.2022.100124.
- Garcia, Andrea, Kumar, Ravi, Huang, Mei, "Whisper AI: Scalable Multilingual Transcription Models," Computational Linguistics Journal, vol. 30, no. 3, pp. 455--470, 2022.
- Aryal, A., Ghahramani, A., Becerik-Gerber, B., and Zhu, Y. (2019). Monitoring and improving workspace environment using IoT and machine learning. Building and Environment, 157, 231-239.
- Hedayati, M., and Dahlan, H. M. (2019). Ergonomic adjustable desk with integrated monitoring system. International Journal of Occupational Safety and Ergonomics, 25(4), 491-503.
- Johnson, D., and Burke, K. (2020). Enhancing workplace ergonomics with smart furniture: A review of technology integration. Applied Ergonomics, 85, 103-112.
- Kim, J., Park, S., and Kang, H. (2019). Development of a smart office desk with personalized features for shared spaces. Journal of Intelligent Manufacturing, 30(5), 1103-1115.

- Chen, Z., Wang, X., and Li, J. (2021). Ergonomic smart desks for workplace wellness: Integrating environmental monitoring and height-adjustment systems. Journal of Workplace Health Management, 14(3), 211-225.
- B. Acar et al., "OPACA: Toward an Open, Language- and Platform-Independent API for Containerized Agents," in IEEE Access, vol. 12, pp. 10012-10022, 2024, doi: 10.1109/ACCESS.2024.3353613.
- C. Rakow, "A framework for simulating mobility services in large scale agent-based transportation systems", Proc. 2019 Summer Simulation Conf., 2019.
- OMG, “Business Process Model and Notation (BPMN), Version 2.0,” Object Management Group, Tech. Rep., 2011. [Online]. Available: http://www.omg.org/spec/BPMN/2.0
- T. Küster, A. Heßler, and S. Albayrak, “Process-oriented modelling, creation, and interpretation of multi-agent systems,” Int. J. Agent-Oriented Software Engineering, vol. 5, no. 2/3, pp. 108–133, 2016.
- Y. Song, W. Xiong, D. Zhu, W. Wu, H. Qian, M. Song, H. Huang, C. Li, K. Wang, R. Yao et al., “RestGPT: Connecting large language models with real-world restful APIs,” arXiv preprint arXiv:2306.06624, 2023.