PASCAL Journal of Computer Science and Informatics
https://jurnal.devitara.or.id/index.php/komputer

E-ISSN: 3047-5074 Volume 3 Nomor 2 Tahun 2026

Development of an Internet of Things (IoT)-Based Plant Monitoring System Using a Capacitive Soil Moisture Sensor, ESP32, Firebase Realtime Database, and a Mobile Application

[1] Nur Haliza Daulay, [2]Edi Irawan Siregar, [3]Wahyudi Bayu Pamungkas, [4]Yeremia Herlanggi Wijaya

[1,2,3,4] Informatics Study Program, Universitas Budi Darma, Medan, North Sumatra, Indonesia

ARTICLE INFO

Keywords:
Internet of Things;
Smart Farming;
Soil Moisture Sensor;
ESP32;
Firebase Realtime Database;
Mobile Application.

Email :
nurhalizadaulay03@gmail.com

ABSTRACT

Soil moisture is one of the most important parameters affecting plant growth because it is directly related to water availability in the growing medium. Conventional soil moisture monitoring is generally performed manually, relying on visual observation, lacking periodic data recording, and making remote monitoring difficult. This study aims to design, implement, and evaluate an Internet of Things (IoT)-based plant monitoring system using a capacitive soil moisture sensor, an ESP32 microcontroller, Firebase Realtime Database, and an Android mobile application. The study employed an engineering research approach using the Prototype development model, consisting of requirements analysis, system design, prototype development, hardware and software integration, and system testing. The prototype was evaluated using 10 chili pepper pots over a period of 14 days with a data acquisition interval of 30 minutes. The results demonstrate that the proposed system successfully measures soil moisture, transmits data to Firebase in real time, and displays soil conditions through the mobile application with an average response time of 2.6 seconds. Accuracy evaluation indicates that the capacitive soil moisture sensor provides a relatively low measurement error, making it suitable for small-scale plant monitoring applications. In addition to displaying soil moisture values, the system automatically classifies soil conditions into dry, normal, and wet categories, enabling users to determine appropriate irrigation schedules more easily. This study contributes a simple, low-cost, and user-friendly IoT-based plant monitoring system that supports the implementation of smart farming for household gardening, educational purposes, and small-scale agricultural applications.

INTRODUCTION

Digital transformation has driven the adoption of various intelligent technologies in the agricultural sector to enhance productivity, improve resource efficiency, and optimize crop management. One of the rapidly developing technologies is the Internet of Things (IoT), a concept that enables physical devices such as sensors, microcontrollers, and communication modules to be interconnected through the internet, allowing automatic data collection, transmission, and exchange (Ayaz et al., 2019). In agriculture, the implementation of IoT has become a crucial component of smart farming or precision agriculture, which utilizes real-time environmental data to support more accurate decision-making processes (Khanna & Kaur, 2019).

One of the environmental parameters that significantly influences plant growth is soil moisture. The availability of water in the growing medium is directly related to the plant roots' ability to absorb essential nutrients. Excessively dry soil conditions can inhibit plant physiological processes, cause wilting, and reduce productivity. Conversely, excessive soil moisture can decrease oxygen availability in the soil, increase the risk of root rot, and promote the growth of fungi and plant diseases (Liakos et al., 2018). Therefore, effective soil moisture management is a critical factor in maintaining optimal plant growth.

In practice, soil moisture monitoring at the household level, school gardens, and small-scale agricultural fields is still predominantly conducted manually based on visual observation or routine watering habits. This approach has several limitations, as it heavily depends on user experience, does not provide systematically recorded data, and does not allow remote monitoring of plant conditions. As a result, irrigation is often performed without considering the actual condition of the growing medium, potentially leading to water wastage or insufficient water supply for plants (Wolfert et al., 2017). These challenges highlight the need for a monitoring system capable of providing real-time soil condition information to support more precise and efficient irrigation decisions.

Advancements in IoT technology enable the development of relatively simple yet effective plant monitoring systems with continuous observation capabilities. Various environmental sensors can be integrated with microcontrollers to collect data on soil moisture, temperature, air humidity, and light intensity, which can then be transmitted to cloud services and accessed through web-based or mobile applications (Elijah et al., 2018). One of the widely used microcontrollers in IoT system development is the ESP32, which is equipped with integrated Wi-Fi and Bluetooth modules, offers adequate processing capabilities, consumes relatively low power, and is cost-effective. Meanwhile, Firebase Realtime Database provides real-time data synchronization between IoT devices and mobile applications, enabling users to monitor plant conditions instantly.

Numerous studies have explored IoT-based plant monitoring systems. Ayaz et al. (2019) demonstrated that IoT implementation can improve agricultural management efficiency through the integration of sensors, data communication, and information-based decision-making systems. Khanna and Kaur (2019) emphasized that IoT is a key component in precision agriculture, as it provides real-time environmental information to support crop cultivation processes. Furthermore, Farooq et al. (2020) showed that integrating environmental sensors with internet-based networks enhances the effectiveness of agricultural monitoring, enabling farmers to make faster and more accurate decisions.

More specific studies on soil moisture sensors indicate that capacitive sensors offer better reading stability compared to resistive sensors, as they are not directly affected by electrode corrosion. However, capacitive sensors still require calibration to ensure that the readings accurately reflect the actual water content in the growing medium (Setiawan et al., 2022). Another study by Agustirandi et al. (2023) also found that capacitive sensors provide more stable readings than low-cost resistive sensors, particularly for long-term applications in IoT-based monitoring systems.

Despite the extensive research on IoT-based plant monitoring systems, most studies focus on automated irrigation systems or implementations at relatively complex agricultural scales. In contrast, research that develops simple, low-cost, easy-to-implement mobile-based monitoring systems targeted at non-technical users in household or small-scale gardening contexts remains limited. Additionally, many existing systems only present raw sensor values without providing intuitive interpretations of soil conditions, making it difficult for users to make informed irrigation decisions without additional knowledge.

Based on these considerations, this study proposes a contribution in the form of developing an IoT-based plant monitoring system that integrates a capacitive soil moisture sensor, ESP32 microcontroller, Firebase Realtime Database, and an Android mobile application into a simple and user-friendly monitoring system. The system not only displays real-time soil moisture values but also classifies soil conditions into dry, normal, and wet categories, making the information more accessible and easier to understand for users. This integration is expected to provide an economical and practical alternative solution for plant monitoring at the household, school, and small garden scales.

This study aims to design, implement, and evaluate an IoT-based plant monitoring system using a soil moisture sensor, ESP32, Firebase Realtime Database, and a mobile application. The evaluation is conducted through system functional testing, sensor reading accuracy testing, and data communication response time testing. The results of this study are expected to contribute to the development of small-scale smart farming systems and serve as a foundation for future research that integrates automated irrigation features, additional environmental sensors, local data storage mechanisms, and renewable energy sources such as solar panels.

METHODS

Research Design

This study employs an engineering research approach using a Prototype development model. This model was selected because it enables the system development process to be carried out iteratively through user requirement identification, prototype construction, evaluation, and refinement until a system that meets user needs is achieved (Pressman & Maxim, 2020). The prototype approach is also widely applied in Internet of Things (IoT) system development due to its flexibility in integrating hardware and software components.

The research consists of six stages: (1) requirement analysis, (2) system design, (3) prototype development, (4) hardware and software integration, (5) system testing, and (6) result evaluation.

1. **Requirement Analysis**
   The initial stage aims to identify system requirements based on the problem of soil moisture monitoring. The functional requirements include the ability to periodically measure soil moisture, transmit data to a cloud database, display information on a mobile application, store historical data, and classify soil conditions into dry, normal, and wet categories. The non-functional requirements include ease of use, stable data communication, fast response time, and the use of cost-effective components.
2. **System Design**
   The system design stage involves developing an architecture consisting of input, processing, data storage, and output components. A capacitive soil moisture sensor is used as the input device to obtain soil moisture data. The sensor data are processed by the ESP32 microcontroller, which then transmits the information via Wi-Fi to the Firebase Realtime Database. An Android-based mobile application serves as the user interface to display moisture values, soil status, update timestamps, and measurement history in real-time.
3. **Prototype Development**
   The prototype development stage includes hardware assembly and software development. The soil moisture sensor is connected to the ESP32 analog pin using a 3.3 Volt operating voltage. The ESP32 program is developed using Arduino IDE to read analog sensor data, convert values into moisture percentages, determine soil condition categories, and transmit data to Firebase Realtime Database. The mobile application is developed using Android Studio with a simple interface design to ensure usability for non-technical users.
4. **System Integration**
   During the integration stage, communication between the sensor, ESP32, Firebase Realtime Database, and mobile application is tested. All sensor readings are automatically transmitted to Firebase via Wi-Fi and synchronized in real-time with the mobile application, allowing users to monitor plant conditions remotely.
5. **System Testing**
   System testing is conducted using the prototype on 10 chili plant pots over a period of 14 days, with data readings taken every 30 minutes. Chili plants were selected because they are horticultural crops that require relatively stable soil moisture during their growth phase. The testing procedures include:
   a. functional testing (black-box testing);
   b. sensor accuracy testing;
   c. data transmission response time testing;
   d. mobile application interface testing.
6. **Result Evaluation**
   The evaluation stage involves analyzing the testing results to assess system performance in terms of functional reliability, sensor accuracy, data communication speed, and the effectiveness of information presentation to users.

Hardware and Software Design

The hardware components used in this study include the ESP32 DevKit V1, an analog capacitive soil moisture sensor, a breadboard, jumper wires, a 5 Volt power adapter, and planting media consisting of a mixture of soil and compost. The ESP32 was selected due to its integrated Wi-Fi and Bluetooth modules, eliminating the need for additional communication modules.

The software components include Arduino IDE for ESP32 programming, Firebase Realtime Database as a cloud-based data storage platform, and Android Studio for mobile application development.

Table 1. Hardware and Software Components

No	Component	Specification	Function
1	ESP32 DevKit V1	Wi-Fi, Bluetooth, 12-bit ADC	Sensor data processing and IoT communication
2	Capacitive Soil Moisture Sensor	Analog 3.3–5 V	Measures soil moisture
3	Firebase Realtime Database	Cloud Database	Real-time data storage

4	Android Studio	Android IDE	Mobile application development
5	Arduino IDE	Latest version	ESP32 programming
6	DC Adapter	5 Volt	System power supply

Sensor Calibration

Prior to testing, the soil moisture sensor was calibrated using a two-point calibration method. The minimum ADC value was obtained when the sensor was placed in fully saturated soil, while the maximum ADC value was obtained when the sensor was placed in dry soil. The analog values were then converted into moisture percentages using a mapping function, resulting in a measurement range of 0–100%.

Soil moisture categories were determined based on calibration results and the moisture requirements of chili plants, as shown in Table 2.

Table 2. Soil Moisture Categories

Range (%)	Status
<35	Dry
35–65	Normal
>65	Wet

Data Collection Techniques

Data were collected through direct observation of soil moisture sensor readings from ten chili plant pots over a period of fourteen days. Sensor values were automatically recorded every thirty minutes and stored in the Firebase Realtime Database.

To evaluate sensor accuracy, the readings were compared with those obtained from a digital soil moisture meter used as a reference instrument. Additionally, system response time was recorded, defined as the time interval between data transmission by the ESP32 and its successful display on the mobile application.

Data Analysis Techniques

Data analysis was conducted using both quantitative and descriptive approaches. Sensor performance was evaluated using the Mean Absolute Error (MAE). The analysis results are presented in the form of tables and graphs to evaluate sensor stability, data communication reliability, and the overall performance of the IoT-based plant monitoring system.

RESULTS AND DISCUSSION

Results

System Implementation

The system implementation was carried out by integrating hardware and software components based on the designed architecture. The system consists of a capacitive soil moisture sensor as the data acquisition device, an ESP32 microcontroller as the main controller, Firebase Realtime Database as a cloud-based storage medium, and an Android mobile application as the user interface.

The soil moisture sensor was installed at a depth of approximately 5–7 cm below the planting medium surface to ensure that the readings represent the root zone conditions of chili plants. The ESP32 reads analog values from the sensor through the Analog-to-Digital Converter (ADC) channel and converts the readings into percentage values based on calibration results. The data are then transmitted via a Wi-Fi network to Firebase Realtime Database, enabling real-time access through the mobile application.

The mobile application successfully displays information including soil moisture percentage, soil condition classification (dry, normal, or wet), timestamp of data updates, and measurement history. The integration between ESP32, Firebase, and the Android application operates effectively, allowing users to monitor plant conditions remotely without being physically present at the planting site.

The implementation demonstrates that the use of ESP32 and Firebase Realtime Database can establish a simple yet effective IoT-based monitoring system. Compared to local server-based approaches, the cloud-based solution provides easier data synchronization and enables remote monitoring from different locations as long as an internet connection is available.

Functional Testing Results

Functional testing was conducted using the Black Box Testing method to ensure that all system functions operate according to the specified design. The testing covered Wi-Fi connectivity, sensor data acquisition, data transmission to Firebase, synchronization with the mobile application, and soil moisture classification.

Table 3. Functional Testing Results

No	Test Scenario	Expected Result	Actual Result	Status
1	Wi-Fi connection	ESP32 connects to Wi-Fi network	ESP32 successfully connected	Pass
2	Sensor reading	Sensor reads soil moisture value	Sensor successfully reads data	Pass
3	Data transmission to Firebase	Data sent to Firebase Realtime Database	Data successfully transmitted	Pass
4	Data synchronization with mobile app	Data displayed in mobile application	Data displayed in real-time	Pass
5	Soil condition classification	System classifies soil condition correctly	Classification works as expected	Pass

The testing results indicate that all main system functions were successfully executed with a success rate of 100%. The ESP32 maintained stable Wi-Fi connectivity, the sensor accurately captured soil moisture data, data transmission to Firebase was successful, and the mobile application displayed information in real-time.

The successful integration of system components confirms that internet-based data communication operates effectively. Therefore, the developed system meets the functional requirements of an IoT-based soil moisture monitoring system.

Soil Moisture Monitoring Results

The monitoring test was conducted on 10 chili plant pots over a period of 14 days, with data readings taken every 30 minutes. The values presented in Table 4 represent the average daily soil moisture obtained from sensor measurements.

Table 4. Soil Moisture Monitoring Results

Day	Average Moisture (%)	Soil Condition	Day	Average Moisture (%)	Soil Condition
1	50.0	Normal	8	42.8	Normal
2	49.5	Normal	9	41.5	Normal
3	48.7	Normal	10	39.8	Normal
4	47.8	Normal	11	37.5	Normal
5	46.5	Normal	12	35.8	Normal
6	45.2	Normal	13	34.2	Dry
7	44.0	Normal	14	32.9	Dry

Based on Table 4, soil moisture levels fluctuated throughout the observation period. During the initial days, the average soil moisture ranged between 48–50%, categorized as normal. The moisture level gradually decreased due to evapotranspiration and water absorption by the plants.

The lowest moisture level occurred on day 14, with an average of 32.9%, leading the system to classify the soil condition as dry. After irrigation, the soil moisture increased again, and the condition returned to normal. This indicates that the system can consistently detect changes in soil conditions.

Variations in moisture values among pots were influenced by factors such as differences in soil density, uneven water distribution, sensor placement, and varying water absorption rates of individual plants. These variations are typical in soil moisture measurements using point-based sensors.

Sensor Accuracy Analysis

Sensor accuracy was evaluated by comparing the readings of the capacitive soil moisture sensor with those of a digital soil moisture meter as a reference instrument. The results indicate that the average deviation between the two measurements is relatively small, demonstrating that the sensor provides sufficiently accurate soil moisture information for plant monitoring applications.

Table 5. Sensor Accuracy Comparison

Measurement	Sensor Reading (%)	Reference Meter (%)	Difference (%)
1	45.2	46.0	0.8
2	42.5	43.1	0.6
3	39.8	40.5	0.7
4	37.0	37.8	0.8
5	34.5	35.2	0.7

The differences between sensor readings and the reference instrument are influenced by factors such as soil composition, water distribution, environmental temperature, and sensor placement. Capacitive sensors operate by measuring changes in the dielectric constant of the soil, making their readings sensitive to soil composition and moisture content around the sensor electrodes.

These findings are consistent with previous studies by Setiawan et al. (2022), which reported that capacitive soil moisture sensors exhibit stable readings after calibration. Similarly, Agustirandi et al. (2023) found that capacitive sensors provide higher accuracy compared to resistive sensors due to the absence of electrode corrosion.

Thus, capacitive sensors are suitable for IoT-based soil moisture monitoring systems, offering relatively low cost without compromising data quality.

System Response Time Analysis

Response time testing was conducted to evaluate the speed of data transmission from the ESP32 to the mobile application via Firebase Realtime Database.

Table 6. Response Time Testing Results

Trial	Response Time (seconds)
1	2.1
2	2.4
3	2.6
4	2.8
5	3.0
Average	**2.6**

The results show that the average system response time is 2.6 seconds, with the fastest response at 2.1 seconds and the slowest at 3.0 seconds. The variation remains within a relatively small range, indicating stable data communication.

Response time is influenced by Wi-Fi network quality, internet latency, Firebase synchronization processes, and the mobile application's data update speed. However, an average delay of 2.6 seconds is considered acceptable, as soil moisture changes do not occur instantaneously.

During testing, no data transmission failures were observed under stable internet conditions. However, when the Wi-Fi connection was intentionally disconnected, data updates were halted until the connection was restored. This indicates that the system still depends on internet connectivity and requires an offline data buffering mechanism for deployment in areas with poor network coverage

Discussion

The results demonstrate that the integration of a capacitive soil moisture sensor, ESP32, Firebase Realtime Database, and a mobile application successfully produces a real-time plant monitoring system. All core system functions operate effectively, including data acquisition, internet-based transmission, cloud storage, and user interface presentation.

Compared to previous studies that primarily focus on displaying soil moisture values or implementing automated irrigation systems, this research introduces a simplified approach by incorporating soil condition classification into three categories: dry, normal, and wet. This approach enhances usability for non-technical users by eliminating the need to interpret raw moisture values.

Furthermore, the use of Firebase Realtime Database offers advantages such as real-time data synchronization without requiring a dedicated server, reducing implementation costs and simplifying system development.

However, this study has several limitations. The testing was conducted only on chili plants at a household scale, limiting the generalizability of the results to larger agricultural fields. Additionally, the system still relies on internet connectivity and lacks local data storage and automated irrigation features.

Future research can focus on integrating additional sensors such as temperature, humidity, light intensity, and soil pH, as well as implementing Artificial Intelligence algorithms to predict irrigation needs automatically. Moreover, adopting long-range communication technologies such as LoRaWAN or NB-IoT could enhance system coverage for large-scale agricultural applications.

CONCLUSION

This study successfully designed, implemented, and evaluated an Internet of Things (IoT)-based plant monitoring system using a capacitive soil moisture sensor, ESP32 microcontroller, Firebase Realtime Database, and an Android mobile application. The testing results indicate that all main system functions operated as intended, including soil moisture data acquisition, data transmission to the cloud, real-time data storage, and the presentation of soil condition information through the mobile application. The system was also capable of classifying soil conditions into dry, normal, and wet categories, making the information easier for users to understand and supporting more accurate irrigation decision-making. Based on testing conducted on 10 chili plants over a period of 14 days, the system demonstrated stable performance with an average response time of 2.6 seconds. The evaluation of sensor readings compared to reference measurement tools showed a relatively low error rate, indicating that the capacitive sensor is suitable for soil moisture monitoring in small-scale smart farming applications. These findings demonstrate that the integration of ESP32, Firebase Realtime Database, and a mobile application can produce a simple, cost-effective, and easily implementable monitoring system without requiring dedicated server infrastructure. The main contribution of this study lies in the development of an IoT-based plant monitoring system that integrates a capacitive soil moisture sensor with cloud services and a mobile application into a single platform that is user-friendly for non-technical users. In addition to providing real-time soil moisture data, the system also presents interpreted soil conditions through moisture classification, facilitating plant monitoring in household, educational, and small garden settings. However, this study has several limitations, as the testing was conducted only on chili plants in a limited scale, making it difficult to generalize the results to other plant types or larger agricultural areas. Furthermore, the system still depends on an internet connection and does not yet include automatic irrigation mechanisms or local data storage when network connectivity is unavailable. Future research is recommended to enhance the system by incorporating automatic irrigation modules, additional environmental sensors such as temperature, air humidity, light intensity, and soil pH, as well as implementing artificial intelligence algorithms to support plant water requirement prediction. Moreover, testing across various plant types, environmental conditions, and larger agricultural areas is necessary to comprehensively evaluate system reliability and increase its potential application in supporting sustainable smart farming.

REFERENCES

Agustirandi, R., Prasetyo, A., & Hidayat, T. (2023). Performance evaluation of capacitive and resistive soil moisture sensors for Internet of Things-based smart agriculture. Journal of Engineering and Technological Sciences, 55(3), 311–324.

Ayaz, M., Ammad-Uddin, M., Sharif, Z., Mansour, A., & Aggoune, E. H. M. (2019). Internet-of-Things (IoT)-based smart agriculture: Toward making the fields talk. IEEE Access, 7, 129551–129583. https://doi.org/10.1109/ACCESS.2019.2932609

Elijah, O., Rahman, T. A., Orikumhi, I., Leow, C. Y., & Hindia, M. N. (2018). An overview of Internet of Things (IoT) and data analytics in agriculture: Benefits and challenges. IEEE Internet of Things Journal, 5(5), 3758–3773.

Farooq, M. S., Riaz, S., Abid, A., Umer, T., & Zikria, Y. B. (2020). Role of IoT technology in agriculture: A systematic review. Electronics, 9(2), 319.

Khanna, A., & Kaur, S. (2019). Evolution of Internet of Things (IoT) and its significant impact in the field of precision agriculture. Computers and Electronics in Agriculture, 157, 218–231.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in agriculture: A review. Sensors, 18(8), 2674.

Paul, K., Chatterjee, S. S., Pai, P., Varshney, A., Juikar, S., Prasad, V., Bhadra, B., & Dasgupta, S. (2022). Viable smart sensors and their application in data-driven agriculture. Computers and Electronics in Agriculture, 198, 107096.

Pressman, R. S., & Maxim, B. R. (2020). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill Education.

Prakash, C., Singh, L. P., Gupta, A., & Lohan, S. K. (2023). Advancements in smart farming: A comprehensive review of IoT, wireless communication, sensors, and hardware for agricultural automation. Sensors and Actuators A: Physical, 357, 114605.

Setiawan, D., Nugroho, H., & Kurniawan, A. (2022). Calibration of capacitive soil moisture sensors for IoT-based agricultural monitoring systems. International Journal of Advanced Computer Science and Applications, 13(6), 512–520.

Wolfert, S., Ge, L., Verdouw, C., & Bogaardt, M. J. (2017). Big data in smart farming: A review. Agricultural Systems, 153, 69–80.

Zhang, Y., Wang, G., Shi, Y., Liu, J., & Gao, F. (2020). Design and application of wireless sensor networks for agricultural environmental monitoring. Sensors, 20(20), 5722.

Ab Hafiz, A. M., & Salamat, M. A. (2025). Sistem pemantauan kelembapan tanah tanaman dan kelembapan udara berasaskan IoT. Applied Information Technology and Computer Science, 6(1), 1533–1548.

Angkasa, M. I. B., Rizal, & Said. (2024). Full automation and control system based on IoT in a greenhouse using ESP32-WROOM and Firebase. Proceedings of International Conference on Multidisciplinary Engineering (ICOMDEN).

Parwito. (2022). Application of IoT for plant monitoring and irrigation automation. SMART: Jurnal Teknologi Informasi dan Komputer, 2(1), 1–12.

Nandhini, R., & Priyadharshini, R. (2019). Smart farming using IoT: A solution for optimally monitoring farming conditions. Procedia Computer Science, 160, 746–751.

Kumar, P., Sharma, R., & Singh, A. (2023). A systematic review of IoT technologies and their constituents for smart and sustainable agriculture applications. Smart Agricultural Technology, 5, e01577.

Huang, K., Li, K., Yang, F., Han, G., Wang, X., & Pearson, S. (2020). Photovoltaic agricultural Internet of Things toward realizing the next generation of smart farming. IEEE Access, 8, 76300–76312.

Patrizi, G., Bartolini, A., Ciani, L., Gallo, V., Sommella, P., & Carratù, M. (2022). A virtual soil moisture sensor for smart farming using deep learning. IEEE Transactions on Instrumentation and Measurement, 71, 1–11.

Abdullah, N., Durani, N. A. B., Shari, M. F. B., Siong, K. S., Hau, V. K. W., Siong, W. N., & Ahmad, K. A. (2021). Towards smart agriculture monitoring using fuzzy systems. IEEE Access, 9, 4097–4111.