# iLand Web Application

**Author** : Thanyared Chotprasert
**Advisor** : Noparut Vanitchanant, Ph.D.
**Company** : T.C.C. TECHNOLOGY CO., LTD.

## ABSTRACT

iLand web application was developed with the objectives of enabling users to conveniently and systematically access and manage land parcels, buildings, legal documentation, and related tax information. The system was designed to address and alleviate the limitations of the previous document-based workflow, which caused delays in data management and retrieval. The developed features include managing detailed information of land parcels and buildings, storing related documents, generating tax reports at both category and company levels, managing project and category data, and modifying tax rates. In terms of development responsibility, the author was responsible for designing and developing the User Interface of the web application, including integrating it with APIs [1] to ensure accurate data display and interaction. System testing results indicate that the application is fully functional and supports more efficient management of land parcel and building data in alignment with the project objectives.

## INTRODUCTION

The management of land parcel, building, legal documentation, and related tax data plays an essential role in organizational operations, as such data must be accurate and traceable. However, the previous paper-based system resulted in delays in data management and retrieval processes. To alleviate these issues, iLand web application was developed to provide a systematic platform that enables users to efficiently manage land parcel and building data.

## CONCLUSION

From the development of iLand web application, which is designed to manage land, building, legal documentation, and related tax information, the system is capable of supporting systematic data storage and modification. The implemented features include displaying and editing land parcel details, managing building information, uploading related documents, generating tax reports at both category and company levels, managing project and category data, and handling tax rate management. The development results demonstrate that all implemented features function correctly in accordance with the specified system requirements and contribute to improving the efficiency of organizational data management.

## TECHNOLOGY

## METHODOLOGY

Requirement analysis.

↓

Frontend development based on Figma UI design.

↓

Function implementation using TypeScript [2], React [3], and Mantine[4].

↓

API integration.

↓

Testing and improvement.

## REFERENCES

[1] AWS. What is a RESTful API Available: https://aws.amazon.com/th/what-is/restful-api/. Accessed: 4 October 2025.
[2] Pryncwill819. What is TypeScript. Available: https://medium.com/@pryncwill819/what-is-typescript-35f4e6f37bd5. Accessed: 4 October 2025.
[3] Xspaces2011. Introduction to React. Available: https://medium.com/@xspaces2011/introduction-to-react-f197c4eb67f4. Accessed: 3 October 2025.
[4] Mohamed H. Mantine UI: the React component library you need to know about. Medium. Available: https://medium.com/@m074554n/mantine-ui-the-react-component-library-you-need-to-know-about-ff4bcde6c93f. Accessed: 4 October 2025.