



Engineering
& Computing

Knight Foundation School of Computer & Information Sciences

Spring 2026 Senior Design Project

Employee Onboarding and IT Access Management Portal

Students: Horacio Andrade, Amber Garcia, Sebastien Guillen, Ana Gutierrez, Jerry Symbleme

Instructor/Faculty: Masoud Sadjadi, Florida International University

PROBLEM

During the onboarding process, organizations often experience delays and miscommunications between HR, Managers, and IT. Tasks like laptop provisioning, account set-up, and approvals are not tracked in one system, leading to incomplete onboarding tasks, delayed system access, and an overall poor new-hire experience.

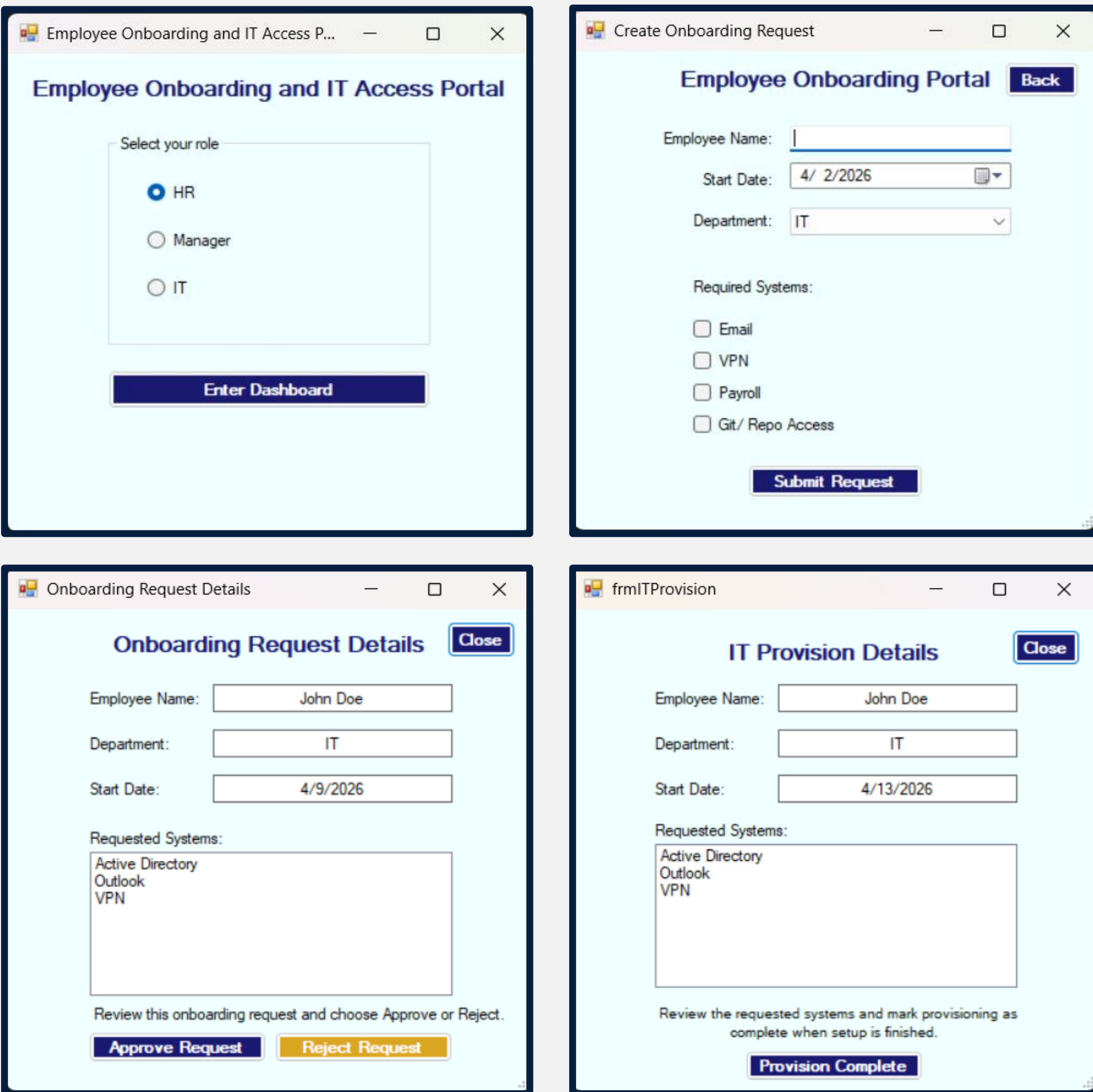
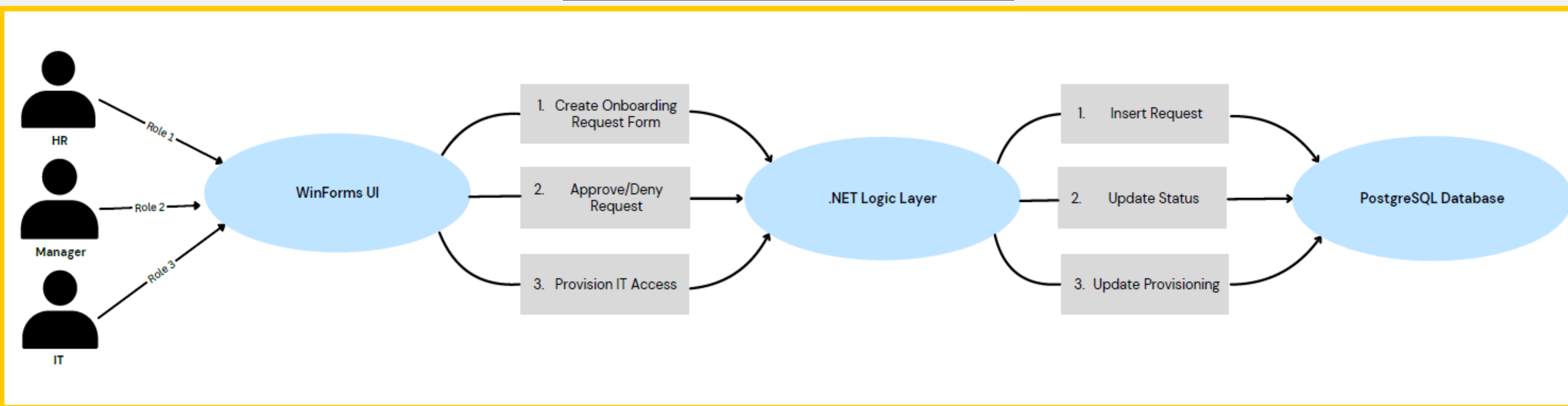
CURRENT SYSTEM

There is no central onboarding progress tracker for HR, Managers, and IT to be in sync. Onboarding tasks are incomplete, delayed, or forgotten. This becomes an issue when the new employee must begin their role and does not have proper access.

OBJECT DESIGN

- The presentation layer is made with WinForms, allowing role-based access and user interfaces and makes forms for HR, Managers, and IT.
- HR staff are the primary users and need to create onboarding requests and track their progress.
- The manager needs to approve or deny the onboarding request.
- The IT technician needs a clear list of provisioning tasks to complete them efficiently.
- The business logic layer is responsible for the authentication, request, approval, and provisioning services.
- The data access layer connects to the PostgreSQL database, which holds SQL queries and parameterized commands.

SYSTEM DESIGN



IMPLEMENTATION

- N-Tier / Layered Architecture is utilized to structure code in logical layers—presentation, business, and data-access.
- Portal is built as a WinForms desktop application that allows users to interact with the onboarding system through a graphical interface.
- Business logic handles the processing of requests such as updating statuses, assigning tasks, and retrieving information from the database.
- Connects to a database where all the data is stored, including employees, onboarding requests, and IT tasks.
- All parts work together so that user interface, business logic, and database support the onboarding workflow.

REQUIREMENTS

Functional Requirements

- HR can create onboarding requests.
- Managers can approve and reject onboarding requests.
- IT can view approved requests.
- IT can submit provisioning updates.
- System tracks onboarding status.
- Role-based access control (HR, Manager, IT).

Non-functional Requirements

- Desktop-based application in a Windows environment
- PostgreSQL relational integrity
- Response time less than 2 seconds for standard queries
- Data persistence and backup
- Version-controlled via GitHub
- Maintainable layered architecture

VERIFICATION

- The implemented security measures help protect the system's functionality and employee information.
- Includes restricted database access, parameterized SQL, and user input validation.
- Role-based access control ensures different users only access features related to their role (HR, manager, or IT).

SUMMARY

The Employee Onboarding and IT Access Management Portal is a desktop application designed to streamline and track the employee onboarding process across HR, Management, and IT departments. Organizations often struggle with the timely passage of information between the different departments when onboarding a new employee. There is no defined onboarding progress tracker, leaving certain onboarding tasks to be delayed or forgotten. Our solution facilitates the transfer of onboarding information between all departments involved via onboarding forms, manager review forms, IT provisioning forms, and an onboarding status tracker. By hosting the information in one application, it is easier for the responsible departments to review onboarding progress and complete the necessary onboarding tasks. The application improves efficiency of employee onboarding, ensuring that new hires get started in their role faster.

REFERENCES

- [1] "PostgreSQL 18.3 Documentation," *PostgreSQL Documentation*, Feb. 26, 2026. <https://www.postgresql.org/docs/current/>
- [2] "Windows Forms for .NET documentation," *Microsoft.com*, 2022. <https://learn.microsoft.com/en-us/dotnet/desktop/winforms/>
- [3] G. Lindemulder and M. Kosinski, "What is role-based access control (RBAC)?," *IBM*, Aug. 20, 2024. <https://www.ibm.com/think/topics/rbac>
- [4] "How to Draw Architecture Diagrams," *GeeksforGeeks*, Mar. 04, 2025. <https://www.geeksforgeeks.org/system-design/how-to-draw-architecture-diagrams/>
- [5] "Layered (N-Tier) Architecture in .NET Core," *DEV Community*, Oct. 08, 2024. <https://dev.to/dotnetfullstackdev/layered-n-tier-architecture-in-net-core-51ic>
- [6] "Adding a workflow status badge - GitHub Docs," *GitHub Docs*, 2026. <https://docs.github.com/en/actions/how-tos/monitor-workflows/add-a-status-badge>
- [7] Jean-Paul Boodhoo, "Design Patterns: Model View Presenter," *learn.microsoft.com*, 2019. <https://learn.microsoft.com/en-us/archive/msdn-magazine/2006/august/design-patterns-model-view-presenter>
- [8] "Separation of Concerns (SoC)," *GeeksforGeeks*, Feb. 13, 2024. <https://www.geeksforgeeks.org/software-engineering/separation-of-concerns-soc/>
- [9] "Basic writing and formatting syntax," *GitHub Docs*. <https://docs.github.com/en/get-started/writing-on-github/getting-started-with-writing-and-formatting-on-github/basic-writing-and-formatting-syntax>

FLORIDA INTERNATIONAL UNIVERSITY

ACKNOWLEDGEMENT The material presented in this poster is based upon the work supported by Florida International University. We thank Masoud Sadjadi for his assistance and mentorship that we received throughout the senior design project.