

SEPTEMBER 24, 2024

APPLIED RESEARCH ON HOW TO IMPLEMENT A SYSTEM TO OPTIMIZE EFFICIENCY AND SECURITY

RESEARCH PAPER

SANCHEZ, FRANCISCO; THOMAS, GILTON; PESTANA, CRISTIANO;
WERNECK ROALE, MIGUEL; LE, MINH

FONTYS UNIVERSITY OF APPLIED SCIENCE
Rachelsmolen 1, 5612MA, Eindhoven

TABLE OF CONTENTS

1. Main Research Question.....	2
2. Sub-Questions	2
3. Research Methodology	3
3.1 Overview of the DOT Framework.....	3
3.2 Research methods.....	3
4. Research Questions and Methods	5

1. MAIN RESEARCH QUESTION

The project aims to develop a full-stack office reservation system that optimizes workspace usage and enhances flexible work arrangements by implementing core features such as room reservations, room management, work scheduling, team management, and administrative control. To determine the optimal way to integrate these functions into a cohesive system, the team has chosen the main research question that is guiding this project:

- **How to implement an office reservation system to optimize efficiency and security while supporting room reservations, work scheduling, team management, and administrative control?**

This question is essential because it will help the team determine the optimal way to integrate these functions into a cohesive and scalable system. By answering this question, the team aims to provide an effective solution that allows the organization to efficiently manage their workspaces and teams.

2. SUB-QUESTIONS

1. How can encryption techniques be employed to protect sensitive information collected?

- Spring Boot (Back-end): Implement encryption methods to secure sensitive data such as personal user information. Ensure data is encrypted both at rest (in the database) and in transit (using HTTPS).
- React (Front-end): Ensure secure transmission of data using HTTPS and handle encryption and decryption processes as needed for any sensitive user data.

2. What methods are there for implementing a real-time notification system?

- Spring Boot (Back-end): Use WebSockets or an event-driven architecture to send real-time notifications about changes, such as new reservations or cancellations.
- React (Front-end): Implement real-time notification features using components like Toast or modal popups to instantly notify users of booking updates or other important information.

3. What measures can be implemented to ensure optimal performance when storing and retrieving data?

- Spring Boot (Back-end): Use efficient data structures and optimize database queries to improve data retrieval performance.
- React (Frontend): Create forms where users can set their work schedule and sync them with the backend.

4. How can continuous integration (CI) and continuous deployment (CD) practices be implemented to automate the testing, build, and deployment processes of the software solution?

- Spring Boot (Back-end): Use GitLab CI to set up CI/CD pipelines that automate the testing, building, and deployment processes of the back end.

- React (Front-end): Automate the deployment of the front-end using CI/CD pipelines, including unit tests and integration tests to ensure reliability at each stage of development.
- 5. How can room availability be maximized and booking conflicts avoided?**
- Spring Boot (Back-end): Implement scheduling algorithms and use database transactions to ensure that room booking conflicts are handled properly, preventing double bookings.
 - React (Front-end): Create a real-time availability checker that dynamically updates to show available rooms and prevent users from selecting rooms already booked by others.
- 6. How can the email system improve communication for employees with different work patterns?**
- Spring Boot (Back-end): Integrate email services like JavaMail or third-party services like SendGrid to automate email notifications and reminders for bookings.
 - React (Front-end): Show in-app notifications and integrate with calendar APIs to remind users of bookings.
- 7. How can single sign-on (SSO) be added to the system?**
- Spring Boot (Back-end): Implement SSO using OAuth 2.0 or OpenID Connect to allow users to securely authenticate on the system using a single login.
 - React (Front-end): Integrate authentication libraries to handle SSO, allowing users to log in with their existing credentials seamlessly across platforms and services.

3. RESEARCH METHODOLOGY

3.1 Overview of the DOT Framework

The research focuses on developing a product rather than creating new scientific knowledge. To structure the research effectively, the team will use the Development Oriented Triangulation (DOT) framework. The DOT framework helps the team to organize the research activities and communicate the findings clearly.

The ICT project requires a combination of various ICT research methods. The team will use the toolkit available at [ICT Research Methods](#) to select appropriate methods for the study. The selected methods are categorized into four main areas: Library, Field, Lab, Showroom, and Workshop. Below is a detailed explanation of each category and the methods chosen.

3.2 Research methods

- **Library**
 - **Community Research:** Community research is using online community knowledge to find answers, check solutions, and share what you learn.
 - **Available Product Analysis:** Available product analysis checks if your project already exists, helping you decide whether to recreate, buy, or use existing solutions.
- **Field**

- **Domain Modelling:** Domain modelling maps out key concepts and their relationships within a specific area to ensure clear understanding, often using techniques like UML class diagrams. It involves consulting stakeholders and developing a conceptual model to analyze the domain before progressing with a project.
- **Explore User Requirements:** Exploring user requirements involves talking to users to understand their needs and how they will use the system. This process includes creating scenarios, use cases, and user stories to capture expected behavior and iteratively refining them with stakeholder feedback.
- **Interview:** Interviews help understand users' opinions, behaviors, goals, and experiences by asking questions in a comfortable setting.
- **Problem Analysis:** Problem analysis involves understanding a problem thoroughly before attempting to solve it. This includes asking questions like who, what, why, when, where, and how, involving domain experts, and ensuring clarity to avoid solving the wrong issue.
- **Lab**
 - **Component Test:** Component testing involves testing a subsystem or component in isolation to ensure it works correctly before integrating it with other parts. This includes checking the component's input and output against expected results using a model, simulation, and evaluation.
 - **System Test:** System testing checks the whole system to ensure it meets its requirements before going live. It involves creating a test plan with expected outcomes, running multiple test rounds, and comparing the results to find and fix bugs.
 - **Unit Test:** Unit testing involves checking individual parts of the code, like methods or functions, in isolation to find bugs early and ensure code stability after changes. This requires defining tests for each part, focusing on important cases, calculations, and exceptions, often using modern IDE tools.
 - **Usability Test:** Usability testing finds and fixes user issues before a system goes live. It uses prototypes or actual solutions to test interfaces, performance, and typical tasks, often with users thinking aloud and using automatic measurement tools.
- **Showroom**
 - **Peer Review:** Peer review improves your work by having colleagues and experts review it in a structured way. This can range from informal reviews to formal sessions, and the key is selecting the right reviewers and providing clear instructions.
 - **Product Review:** A product review ensures the product meets quality standards and requirements before release. This involves a client representative and experts walking through the product, demonstrating all relevant parts, and checking against the requirement list.
- **Workshop**

- **Brainstorm:** Brainstorming generates and develops new ideas by bringing people together to spark creativity. Use creativity techniques, accept all ideas without early criticism, and build upon each other's ideas.
- **Code Review:** Code reviews help improve code quality by having peers identify bugs and suggest improvements. This can be done through team sessions, pair programming, or requiring reviews for each commit, ensuring adherence to coding standards and continuous feedback.
- **IT Architecture Sketching:** IT architecture sketching helps define complex IT architecture by facilitating discussions among software designers and architects. Gather around a whiteboard, sketch high-level designs, and focus on important details only when necessary.
- **Requirements Prioritization:** Requirements prioritization involves defining and ranking the requirements of a new design or redesign by gathering input from stakeholders. This process ensures clear, complete requirements and uses prioritization techniques like MoSCoW to address the most critical needs first, minimizing development risks.

By employing these methods, the team aims to develop a comprehensive and effective software solution that meets the specific needs of an office reservation system.

4. RESEARCH QUESTIONS AND METHODS

- 1. How can encryption techniques be employed to protect sensitive information collected?**
 - Research methods: Community Research, Domain Modelling, Component Test, System Test, Unit Test, Peer Review, Brainstorm, Code Review.
- 2. What methods are there for implementing a real-time notification system?**
 - Research methods: Community Research, Available Product Analysis, Problem Analysis, Component Test, System Test, Unit Test, Usability Test, Peer Review, Brainstorm, Code Review.
- 3. What measures can be implemented to ensure optimal performance when storing and retrieving data?**
 - Research methods: Community Research, Domain Modelling, Problem Analysis, Component Test, System Test, Unit Test, Peer Review, Code Review, IT Architecture Prioritization.
- 4. How can continuous integration (CI) and continuous deployment (CD) practices be implemented to automate the testing, build, and deployment processes of the software solution?**
 - Research methods: Community Research, Explore User Requirements, Brainstorm.
- 5. How can room availability be maximized and booking conflicts avoided?**

- Research methods: Domain Modelling, Explore User Requirements, Problem Analysis, Component Test, System Test, Unit Test, Usability Test, Peer Review, Product Review, Brainstorm, Code Review, IT Architecture Sketching, Requirements Prioritization.
- 6. How can the email system improve communication for employees with different work patterns?**
- Research methods: Community Research, Available Product Analysis, Explore User Requirements, Interview, Problem Analysis, Brainstorm.
- 7. How can single sign-on (SSO) be added to the system?**
- Research methods: Community Research, Available Product Analysis, Explore User Requirements, Interview, Brainstorm.