ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Information Systems Analysis & Design · Group 13 · 2023–2024

WorkPortal

HR Employee Management System

Complete analysis, design, and implementation of a web-based HR portal for G4S Greece, grounded in the Soft Systems Methodology (SSM). From problem identification to a cloud-deployed application.

Course:	Information Systems Analysis & Design
Team:	Group 13
University:	Athens University of Economics and Business (AUEB)
Live Demo:	workportal.koyeb.app

Abstract

This report presents the complete analysis, design, and implementation of WorkPortal – a web-based HR portal for human resource management, developed as a practical application within the study of information systems. The work is grounded in a real organizational problem identified at G4S Greece, the country's leading private security company, through interviews with key stakeholders.

The Soft Systems Methodology (SSM) was applied to diagnose a chronic communication failure between frontline security guards and the HR/IT departments. The problem was traced to a one-way messaging architecture with no feedback channel from employees. Five root definitions were developed through CATWOE analysis, producing a formal requirements model with ten use cases.

WorkPortal resolves this problem with a bidirectional portal built on Java 17, Jakarta Servlets & JSP, SQLite, and Apache Tomcat 10.1 – containerized with Docker and live on Koyeb. Employees submit complaints, leave requests, and meeting requests directly to HR; responses are returned to personal inboxes and the schedule view.

Application type	Web-based HR self-service portal (Jakarta Servlets & JSP)
Backend	Java 17 · Apache Tomcat 10.1
Database	SQLite (embedded, xerial JDBC driver)
Methodology	Soft Systems Methodology (SSM) – Checkland
Hosting	Koyeb · Frankfurt · free tier
Live URL	workportal.koyeb.app

Keywords: Soft Systems Methodology, CATWOE, HR Information Systems, Java Servlets, JSP, SQLite, Docker, Employee Self-Service

Contents

1 Introduction

1.1 Background – G4S Greece

G4S (Group 4 Securicor) is a British multinational security services company headquartered in London, with revenues exceeding $20 billion and a global workforce of more than 800,000 employees. G4S Greece entered the market in 1978 as Group 4 Securitas Hellas and remains the country's leading private security company. In 2021, it was acquired by Allied Universal®.

The Greek subsidiary employs approximately 2,800 people: roughly 2,400 security guards and supervisors stationed at client facilities across the country, 150 middle-management supervisors, and 250 head-office staff.

1.2 The Communication Problem

Through interviews with the IT Director, the HR Director, the IT Project Manager, and the On Call Rostering unit, a chronic structural problem emerged: G4S Greece relied on a one-way messaging system – instructions flowed only from head office downward, with no formal reverse channel.

Core problem: Security guards stationed at client facilities cannot communicate directly with HR or IT. Every request – for leave, complaints, or meetings – passes through supervisors, causing multi-day delays and information distortion.

1.3 Proposed Solution

WorkPortal is a web-based self-service HR portal that gives every G4S employee a personal corporate account. Through it, employees submit requests directly to HR and receive structured, trackable responses – completely bypassing the relay chain.

Figure 1 contrasts the legacy communication flow with the flow enabled by WorkPortal.

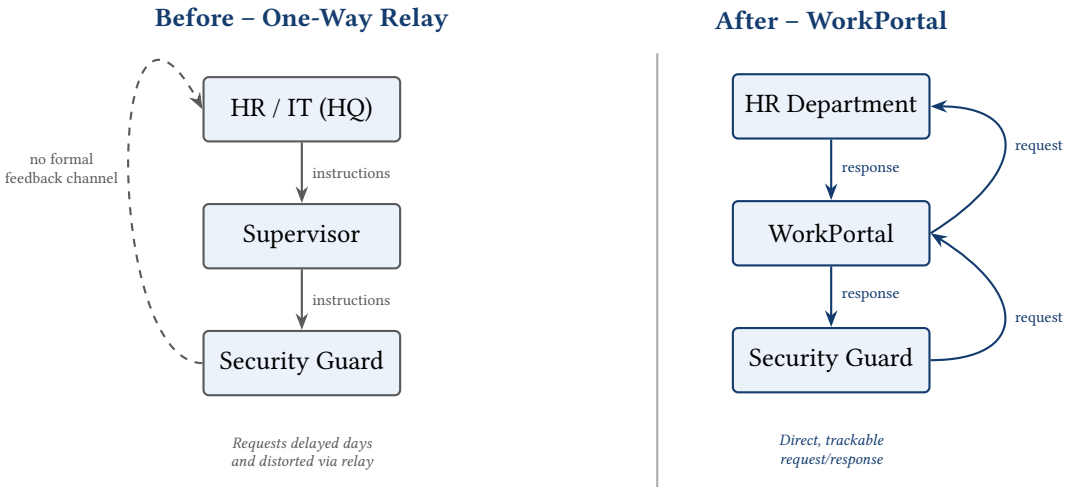

Figure 1: Communication flow before and after WorkPortal

2 Methodology

2.1 Soft Systems Methodology (SSM)

The Soft Systems Methodology (SSM) of Peter Checkland (Lancaster University) was applied. SSM is suited to ill-structured problems in which stakeholders hold conflicting worldviews. Rather than presupposing a solution, SSM records each viewpoint, formulates it as a root definition, builds a conceptual model, and identifies feasible changes.

Table 1: Stages of the SSM process

#	Stage	Description
1	Problem Expression	Rich picture via stakeholder interviews
2	Stakeholder Mapping	Identification of actors and worldviews
3	Root Definitions (CATWOE)	Formulation of each viewpoint
4	Conceptual Model	Derivation of ideal system activities
5	Comparison	Mapping the model against current reality
6	Feasible Changes	Definition of desirable and implementable changes

2.2 Organizational Hierarchy

Table 2: Organizational hierarchy at G4S Greece

Level	Role / Comment
Head Office Departments	HR · IT · On Call Rostering
Middle Management	150 supervisors
Site Supervisors	Communication relay (organizational overhead)
Frontline Guards	≈2,400 · primary WorkPortal users

2.3 Stakeholder Analysis

Table 3: Stakeholder analysis

Stakeholder	Role / Interest	Problem
Security Guards	Frontline employees · primary users	No direct channel to HR
Site Supervisors	Relay communication upward	Burdened with administrative overhead
Middle Managers	Middle management · information hub	Congestion without real authority
HR Department	Labor relations, approvals, scheduling	No direct access to the frontline
IT Department	Infrastructure · account management	Manual account creation

2.4 CATWOE Analysis & Root Definitions

Table 4: Root definitions per stakeholder

Stakeholder	Root Definition
Security Guards	A system for direct, unmediated communication in which IT and HR optimize guard-department interaction, contributing to improved job satisfaction.
Site Supervisors	A system that eliminates unnecessary communication overhead so that supervisors can focus on operational oversight rather than relaying information.
HR Department	A system for faster service through automated request handling, enabling HR to gain deeper insight into employees.
IT Department	A system enabling real-time notifications, frontline feedback collection, and corporate account management – saving IT time.
Site Supervisor	A system that removes unnecessary communication responsibility from the supervisor, making them more effective with field staff.

2.5 Conceptual Model Activities

Table 5: Conceptual model activities

#	Activity
1	Survey employees on communication problems
2	Relay structured feedback to HR
3	Process information and build employee profiles
4	Design system architecture and data model
5	Develop and test the portal
6	Create corporate accounts for all employees
7	Enable fast, direct request submission from the frontline
8	Guarantee timely HR responses to every request
9	Distribute corporate news and schedule updates through the portal

2.6 Performance Measurement Criteria

Table 6: Performance measurement criteria

Criterion	Metric
Effectiveness	Reduction in average request response time
Efficiency	Reduction in HR administrative overhead per request
Outcome	Measurable increase in employee satisfaction and retention

3 Requirements Analysis

G4S Greece requires a bidirectional, web-based HR communication portal that allows frontline employees to submit requests, complaints, and meeting proposals directly to HR – and allows HR to respond in a trackable manner – without routing through a supervisor.

3.1 Use Case Model

Figure 2 presents the use case diagram, showing the three actors (Employee, HR, IT) and their associated use cases.

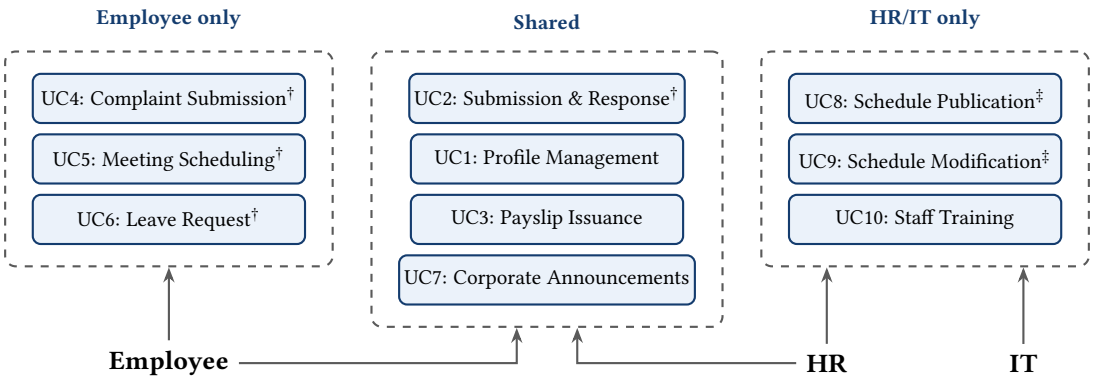

Figure 2: Use case diagram for WorkPortal – grouped by actor. Note: UC1 and UC7 are also served by IT.

Table 7: Use case model

#	Use Case	Actor(s)	Description
UC1	Profile Management	Employee, HR, IT	Creation and update of corporate profiles
UC2	Submission & Response	Employee, HR	Core bidirectional request/response flow
UC3	Payslip Issuance	Employee, HR	Request and retrieval of a digital payslip
UC4	Complaint Submission	Employee	Submission of a formal complaint to HR
UC5	Meeting Scheduling	Employee	Meeting request with participant selection
UC6	Leave Request	Employee	Submission of a leave / day-off request
UC7	Corporate Announcements	Employee, IT	Corporate news on the homepage
UC8	Schedule Publication	HR	Publication of the weekly schedule
UC9	Schedule Modification	HR	Amendment of a published schedule
UC10	Staff Training	HR, IT	Management of training sessions

Relationships: UC2 *includes* UC4, UC5, UC6. UC8 *is extended by* UC9.

3.2 Non-Functional Requirements

Table 8: Non-functional requirements

Characteristic	Requirement
Usability	Usable by employees with minimal digital experience
Reliability	Continuous availability · no scheduled downtime
Performance	Page response within 2 seconds under normal load
GDPR	All personal data compliant with EU Regulation 2016/679
Budget	Development cost €60,000–100,000 · 6–9 month timeline

4 System Design

4.1 Three-Tier Architecture

Figure 3 illustrates the three-tier architecture of WorkPortal: presentation, business logic, and data.

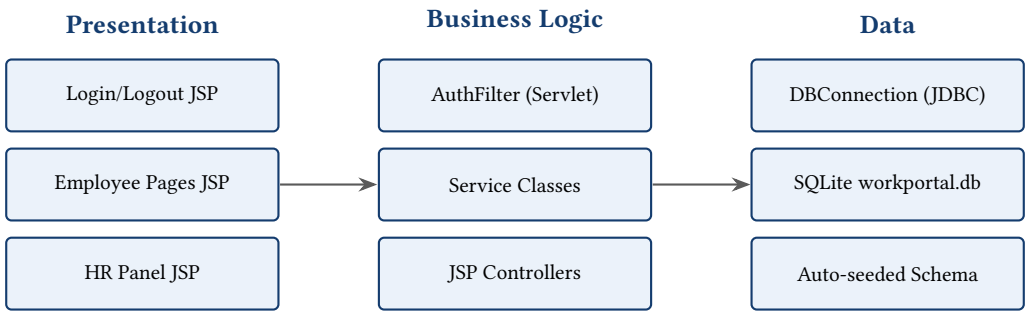

Figure 3: Three-tier architecture of WorkPortal

4.2 Domain Class Model

Figure 4 presents the UML class diagram, showing the four domain entities and their relationships.

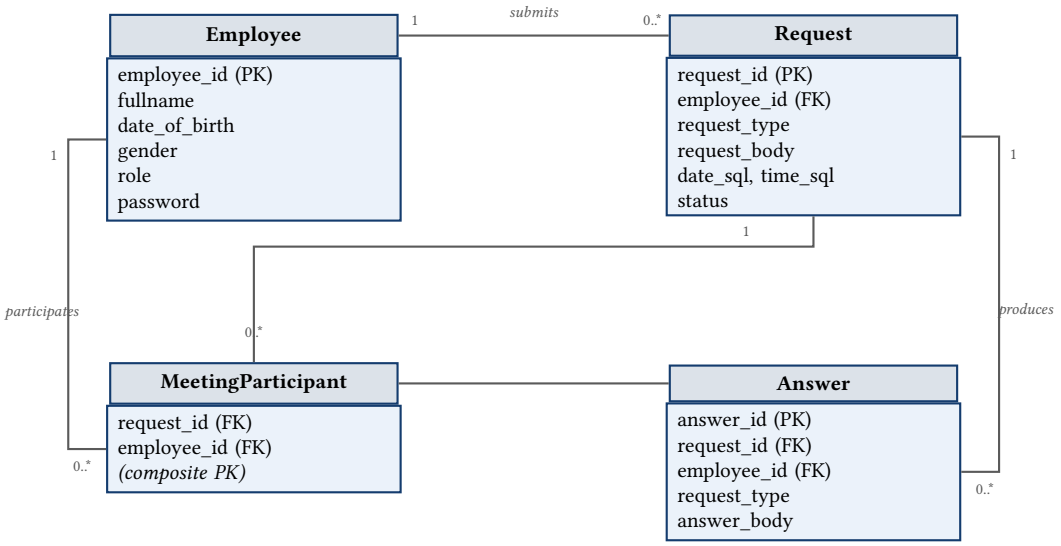

Figure 4: Domain class diagram for WorkPortal

Table 9: Domain class model

Entity	Key Fields	Relationships
Employee	employee_id (PK), fullname, date_of_birth, gender, role, password	Submits 1..* Requests · participates in Meetings
Request	request_id (PK), employee_id (FK), request_type, request_body, date_sql, time_sql, status	Produces 0..* Answers · has 0..* MeetingParticipants
Answer	answer_id (PK), request_id (FK), employee_id (FK), request_type, answer_body	Belongs to one Request
MeetingParticipant	request_id (FK), employee_id (FK) – composite PK	Links Request ↔ Employee for meetings

4.3 Database Schema

Listing 1: SQLite schema

```
CREATE TABLE employees (
 employee_id TEXT PRIMARY KEY, fullname TEXT NOT NULL,
 date_of_birth TEXT NOT NULL, gender TEXT NOT NULL,
 role TEXT NOT NULL, password TEXT NOT NULL
);
CREATE TABLE hr_users (username TEXT PRIMARY KEY, password TEXT NOT NULL);
CREATE TABLE requests (
```

```
 request_id TEXT PRIMARY KEY,
 employee_id TEXT NOT NULL REFERENCES employees,
 request_type TEXT NOT NULL, -- 'complaint' | 'meeting' | 'dayoff'
 request_body TEXT NOT NULL,
 date_sql TEXT, time_sql TEXT,
 status TEXT NOT NULL DEFAULT 'pending'
);
CREATE TABLE meeting_participants (
 request_id TEXT NOT NULL REFERENCES requests,
 employee_id TEXT NOT NULL REFERENCES employees,
 PRIMARY KEY (request_id, employee_id)
);
CREATE TABLE answers (
 answer_id TEXT PRIMARY KEY,
 request_id TEXT NOT NULL REFERENCES requests,
 employee_id TEXT NOT NULL REFERENCES employees,
 request_type TEXT NOT NULL, answer_body TEXT NOT NULL
);
```

4.4 Security

Table 10: Security controls

Control	Implementation
Authentication	Session-based (Jakarta Servlet). Employee: Employee_obj · HR: HR flag. Expires after 30 minutes.
Access Control	AuthFilter (`@WebFilter("/*")`) enforces a role check on every URL.
SQL Injection	All queries exclusively use PreparedStatements.
XSS	All JSP outputs are processed through escapeHtml.jsp.

△ **Known limitation:** passwords are stored in plaintext. Production deployment must use bcrypt or Argon2 hashing.

5 Technical Implementation

5.1 Technology Stack

Table 11: Technology stack

Layer	Technology	Notes
Backend	Java 17 (LTS)	Jakarta EE namespace
Web Framework	Jakarta Servlets + JSP	Servlet API 6.0
App Server	Apache Tomcat 10.1	temurin-jammy base image
Database	SQLite (xerial JDBC)	3.45.1.0 – embedded
Frontend	HTML5, CSS3, Vanilla JS	10 modular stylesheets
Build	Apache Maven 3.9	Packaged as ROOT.war
Container	Docker	Multi-stage Alpine + Tomcat
Cloud	Koyeb	Free tier · persistent volume

5.2 Key Implementation Patterns

Thread-Safe Schema Initialization

DBConnection applies double-checked locking so that schema initialization runs exactly once across all concurrent requests:

Listing 2: DBConnection.java

```
private static volatile boolean initialized = false;
private static final Object LOCK = new Object();

public static Connection getConnection() throws SQLException {
    Connection conn = DriverManager.getConnection(URL);
    try (Statement st = conn.createStatement()) {
        st.execute("PRAGMA journal_mode=WAL");
        st.execute("PRAGMA foreign_keys=ON");
    }
    if (!initialized) {
        synchronized (LOCK) {
            if (!initialized) { // double-check
                initSchema(conn);
                initialized = true;
            }
        }
    }
    return conn;
}
```

Transactional Answer Persistence

HR responses update two tables atomically – failure rolls back both:

Listing 3: AnswerService.java

```
conn.setAutoCommit(false);
try {
    PreparedStatement ins = conn.prepareStatement(
```

```
        "INSERT INTO answers VALUES (?,?,?,?,?)");
    PreparedStatement upd = conn.prepareStatement(
        "UPDATE requests SET status=? WHERE request_id=?");
    conn.commit();
} catch (SQLException ex) {
    conn.rollback();
    throw ex;
}
```

5.3 Application Pages

Table 12: Application pages

Role	Page	Purpose
Both	login / hrLogin	Role-based authentication
Employee	menu.jsp	Homepage with corporate news
Employee	expressComplaint.jsp	Submit a complaint
Employee	arrangeDayOff.jsp	Request leave
Employee	scheduleMeeting.jsp	Schedule a meeting
Employee	answers.jsp	Inbox – HR responses
Employee	schedule.jsp	Schedule of accepted requests
HR	complaints.jsp	View and respond to complaints
HR	meetings.jsp	Accept/reject meetings
HR	daysoff.jsp	Accept/reject leave requests

6 Deployment and Installation

6.1 Docker Multi-Stage Build

A two-stage Dockerfile separates the build-time tools (Maven + JDK) from the runtime image (Tomcat + JRE only), reducing the image size from ≈600 MB to ≈220 MB.

Listing 4: Dockerfile

```
# Stage 1: Build -- Maven + JDK 17
FROM maven:3.9-eclipse-temurin-17-alpine AS build
WORKDIR /app
COPY pom.xml .
RUN mvn dependency:go-offline -q
COPY src/ src/
RUN mvn package -q -DskipTests   # produces ROOT.war

# Stage 2: Runtime -- Tomcat + JRE only
FROM tomcat:10.1-jre17-temurin-jammy
RUN rm -rf /usr/local/tomcat/webapps/*
COPY --from=build /app/target/ROOT.war /usr/local/tomcat/webapps/ROOT.war
RUN mkdir -p /data && chmod 777 /data
EXPOSE 8080
```

```
ENV JAVA_OPTS="-Xms48m -Xmx180m -XX:+UseSerialGC -XX:MaxMetaspaceSize=80m"
CMD ["catalina.sh", "run"]
```

6.2 JVM Memory Optimization

Table 13: JVM memory optimization flags

Flag	Purpose
`-Xms48m / -Xmx180m`	Initial and maximum heap
`-XX:+UseSerialGC`	Minimal GC overhead for low concurrent usage
`-XX:MaxMetaspace-Size=80m`	Class metadata region limit
`-XX:CompressedClass-SpaceSize=32m`	Compressed class pointer space limit

6.3 Cloud – Koyeb

Table 14: Koyeb deployment parameters

Parameter	Value
Region	Frankfurt (EU) – free tier (512 MB RAM, 0.1 vCPU)
Persistence	Koyeb volume at /data – SQLite persists across restarts
URL	`running-jacintha-workportal-b16a6945.koyeb.app`

7 Evaluation

7.1 Requirements Coverage

Table 15: Requirements coverage by use case

#	Use Case	Status	Notes
UC1	Profile Management	✓	Profile view page
UC2	Submission & Response	✓	Complete bidirectional flow
UC3	Payslip Issuance	~	Modeled · not implemented
UC4	Complaint Submission	✓	Complete flow
UC5	Meeting Scheduling	✓	With participant selection
UC6	Leave Request	✓	Complete approval flow
UC7	Corporate Announcements	✓	Employee homepage
UC8	Schedule Publication	✓	Via request acceptance
UC9	Schedule Modification	~	Only via approval flow
UC10	Staff Training	×	Out of scope for v1

7.2 Known Limitations

- Passwords stored in plaintext – bcrypt/Argon2 required for production
- SQLite unsuitable for high concurrent writes – migration to PostgreSQL needed
- No email notifications – employees must log in to check for updates
- No account management UI – accounts are created programmatically
- UC3 (payslip) and UC10 (training) were not implemented in v1

8 Conclusions and Future Work

8.1 Conclusions

This project delivered both a rigorous organizational analysis and a functional, cloud-deployed system that directly resolves the chronic communication failure at G4S Greece. SSM proved well-suited to the task: capturing five distinct stakeholder worldviews through CATWOE produced a coherent set of requirements that guided every design decision in WorkPortal. The use-case model and class diagram map directly onto the implementation.

WorkPortal demonstrates that a lean stack – Java Servlets, JSP, SQLite, Docker – can deliver a quality HR portal deployable on a free-tier cloud (≈150 MB). Most importantly, frontline employees now submit requests and receive structured HR responses without an intermediary.

8.2 Future Development Roadmap

Table 16: Future development roadmap

Phase	Focus	Key Deliverables
1	Security	Password hashing (bcrypt) · CSRF tokens · HTTPS
2	Database	SQLite → PostgreSQL for production scale
3	Missing Features	Payslips (UC3) · Training (UC10)
4	Notifications	Email/push for request status changes
5	Mobile App	Native Android/iOS for field employees
6	Analytics	HR reporting · response times · trends

Appendix

A.1 Interview Sources

Table 17: Interview sources

Name	Role
Christos Ntiskos	IT Director, Central and Southern Europe
Ilias Chondros	HR Director
Foivos Pappas	IT Project Manager
Christina Kapou	On Call Rostering

A.2 Demo Credentials

Table 18: Demo credentials

Role	Portal ID	Password
Employee	E0000001	spilios
HR	HR	123

A.3 Glossary

Table 19: Glossary

Term	Definition
SSM	Soft Systems Methodology – problem analysis for ill-structured human activity situations (Checkland, 1981)
CATWOE	Customers, Actors, Transformation, Worldview, Owner, Environment – an SSM mnemonic for constructing root definitions
One-Way Communication	A top-down-only communication model with no feedback channel from employees
On Call Rostering	HR sub-unit responsible for compiling the weekly work schedule
WAR	Web Application Archive – standard packaging format for Java web apps on Tomcat
WAL	Write-Ahead Logging – an SQLite journal mode that improves concurrent read performance
GDPR	EU General Data Protection Regulation (2016/679)
Jakarta EE	Open-source successor to Java EE, managed by the Eclipse Foundation