INFORMATION MANAGEMENT & SYSTEMS ENGINEERING

MILESTONE 1 SPECIFICATION REPORT

Table of Content

Business Model Outline

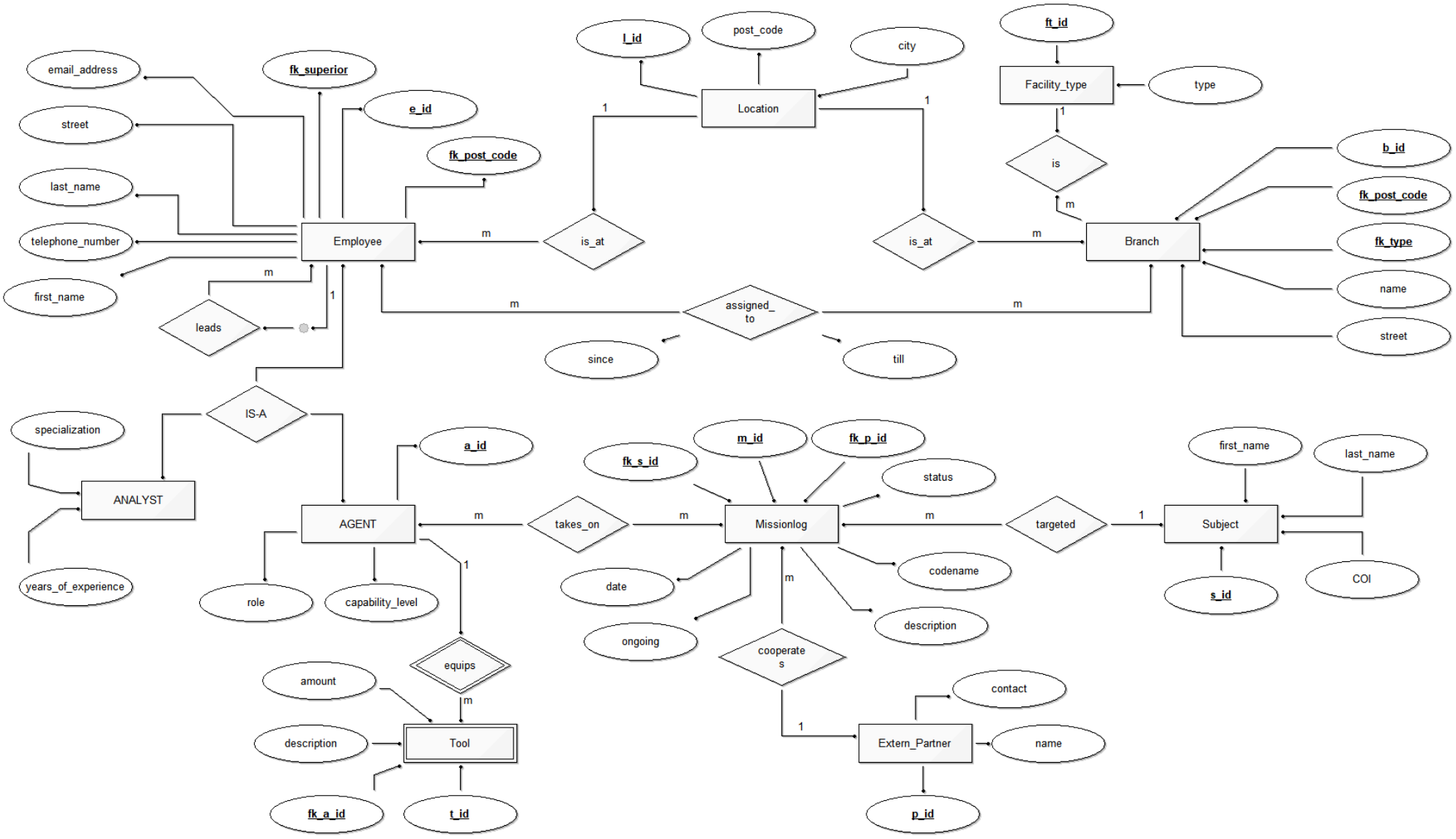
The goal of this project is to create an espionage agency management platform for its administrational departments and operational mission logs. At its core, the platform aims to store and provide data on the agency's employees and their equipment, branches and work places, as well as keeping track on mission logs of employees for their subjects and in cooperation with external partners.

The bureau employs several people with various occupations. Each employee holds personal information, like their name and telephone number, and are identified by an ID. Employees are assigned to one branch of the agency. However, employees are allowed to switch to another branch, so the time period they worked at a certain branch is being tracked off. The branch itself has a name and a street, while also having a facility type to describe what the branch does (Office, Headquarters, …). Both employee and a branch have a location where they reside, which is saved through a post code and the city they are in. Every employee has a higher-up, a supervisor that leads an employee.

Each employee is classified into two possible occupations, those being "Analyst" and "Agent". Analysts are experts in their chosen specialisation and have a certain amount of experience measured in years. Agents have a certain role they perform, a internal grading that measures their skill level, which is called capability level and a unique identifier which is used for mission logs. Each agent equips his own set of tools depending on his role, which can range from one toolkit to several. There is a description for each toolkit as well as a number that showcases how many instances of a tool are currently equipped by the agent.

The bureau can send agents on one or more missions, for which each mission is saved as a mission log. Each mission log has a codename and a description to describe what the goal of the mission is. Furthermore the status of the mission is tracked by showing whether its successful, ongoing or unsuccessful. The log shows whether a mission is currently active or has been already complete, and has a date attached, which says when the mission was carried out. A mission itself has the purpose of tracking which agent(s) currently target which subject of interest. The subject has a first name, last name and the reason why the subject is investigated (COI). A mission can also be carried out in cooperation with a external partner. An external partner can cooperate in one or more missions and is identified with an ID, as well as has a contact and a name.

Entity Relationship Model



Use-Cases

Main Use Case 1: Assign Mission to Agent (LOG1CBOMB)

Objective	Assign mission to agent
Description	The user wants to assign a new or ongoing mission to an agent.
Precondition	- User is connected to the database - User is logged in - At least one agent is available - A mission log exists with a subject under investigation
Expected Execution	- User navigates to mission list - System provides list of mission logs consisting of newly added and ongoing missions - User selects one mission log - System provides list of available agents - User select one or more agents to the previously selected mission - User approves the selection
Postcondition (Success)	- The platform confirms the changes are successful. - The mission has a new agent assigned to it. - The database gets a write command and the changes are updated on the database.
Postcondition (Error)	- The platform throws an error of an unsuccessful assignment. - It does not display a newly assigned agent to the selected mission. - The database does not experience any changes. - The user can navigate through the platform again.

Main Use Case 2: Register employee to branch (QuirkyCroissant)

Objective	Employee gets linked to branch
Description	An employee gets assigned to a specific branch by specific information
Precondition	- Employee already exists in the system - Branch already apparent in system - ‘Entrance date’ has to be known
Expected Execution	1. User navigates to a specific employee on the list and selects it 2. User navigates to the dedicated section for “assignment history” and clicks on an add button 3. User opens designated interface frontend and fills it out 3.1. User selects specific branch 3.2. User adds the “entrance date” that the employee joins the branch 3.3. User submits operation by clicking dedicated button 4. Update list of “assignment history” after inserting data
Postcondition (Success)	- New entry in “assigned_to” gets created - User gets notified at successful insert - “assignment history”-list in the employees profile gets updated
Postcondition (Error)	- Error message gets displayed if user provides system with faulty input and doesn’t continue with the insertion of the new data

Use Case 3: Issue Equipment to Agent (LOG1CBOMB)

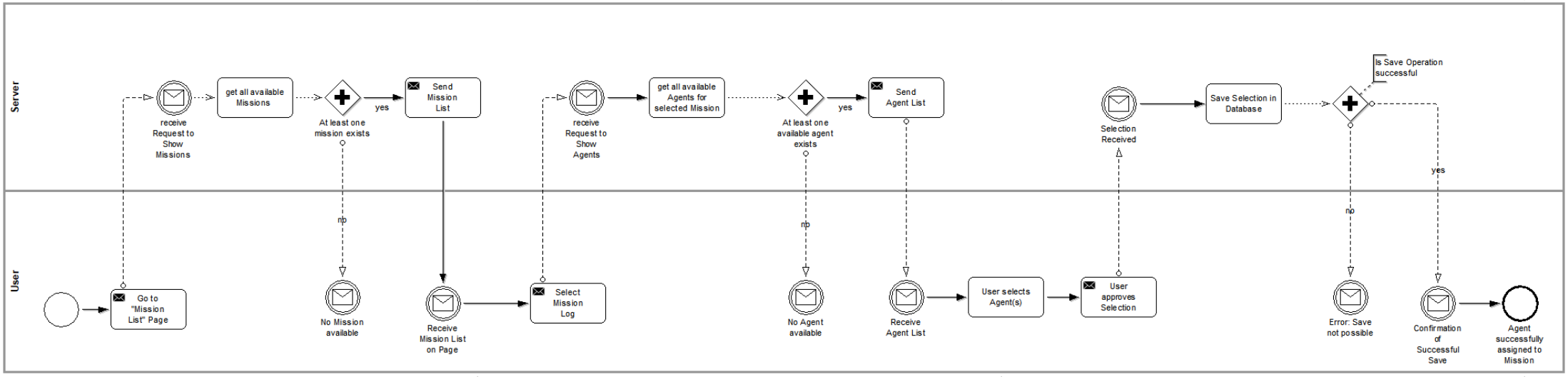
Objective	Issue equipment to agent
Description	The user wants to issue equipment (toolkit) to an existing agent
Precondition	- User is connected to the database - User is logged in - At least one agent exists in the database
Expected Execution	- User navigates to agent list - System provides list of agents with their roles and capability levels - User selects an agent - The user chooses to edit the agent - The user chooses a toolkit for the agent - The user approves his choice of toolkit for the agent - The user approves the edit
Postcondition (Success)	- The platform confirms the changes as successful. - The toolkit has been assigned to the agent and the agent now equips the toolkit - The database gets a write-command and the changes are updated on the database.
Postcondition (Error)	- The platform throws an error of an unsuccessfully issuing the equipment to the agent - It does not display the issued equipment of the agent - The database does not experience any changes. - The user can navigate through the platform again.

Use case 4: Promoting an employee (QuirkyCroissant)

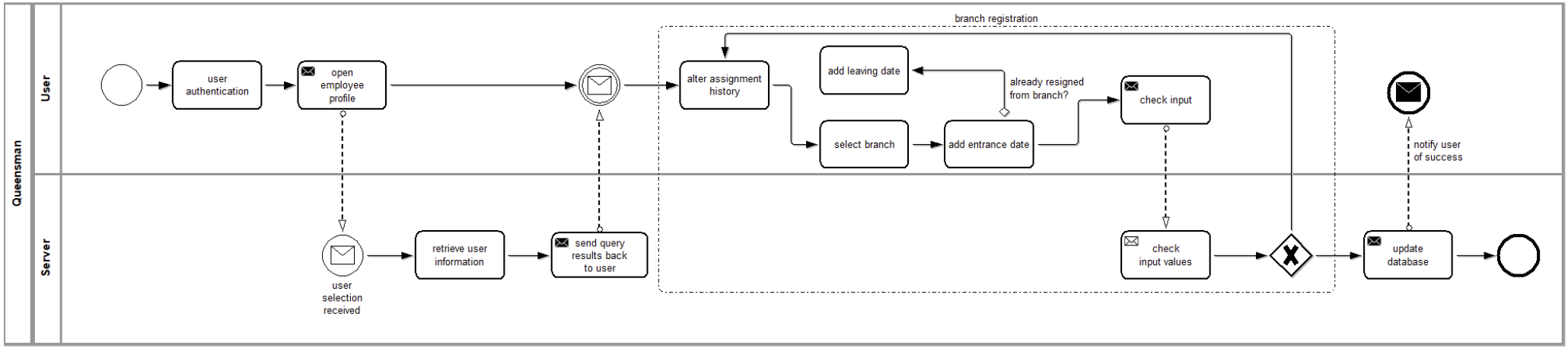
Objective	Employee gets promoted
Description	An employee gets selected to “lead” other employees in the agency
Precondition	- Designated Employee already exists in the system - Multiple employees have to exist in the system
Expected Execution	1. User navigates to a specific employee on the list and selects it 2. User navigates “manage position” submenu and opens designated interface frontend 2.1. User selects specific employees 2.2. User submits operation by clicking dedicated button 3. Update submenu of “manage position”
Postcondition (Success)	- Updating selected employees supervisor attribute pointing to the designated employee, who is now their supervisor - User gets notified at successful update - “Manage position”-submenu in the employees profile gets updated
Postcondition (Error)	- No subordinate selection, error message gets thrown

BPMN diagrams

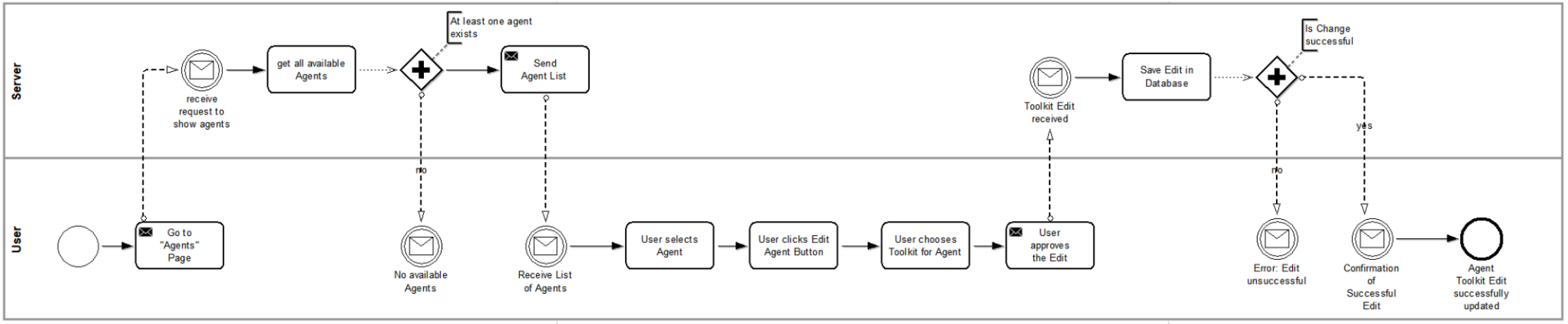
Main Use Case 1: Assign Mission to Agent (LOG1CBOMB)



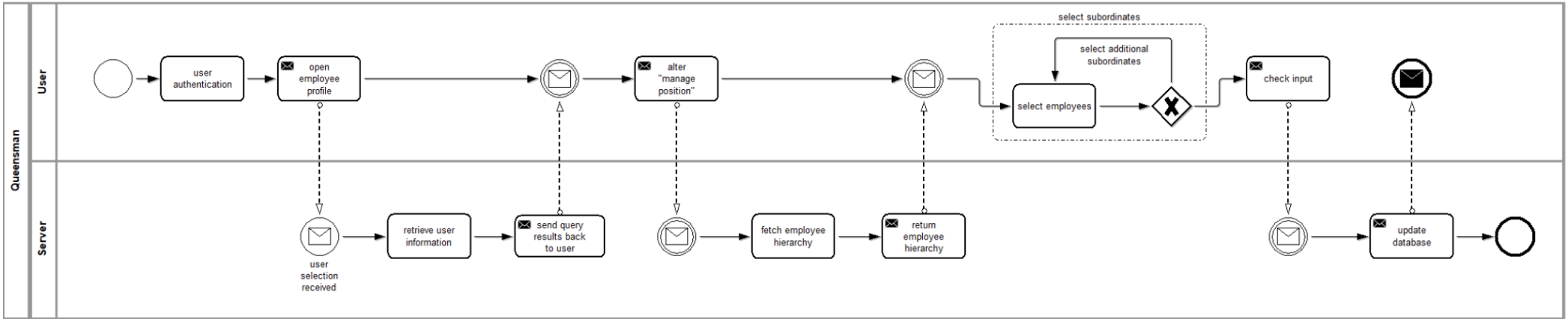
Main Use Case 2: Register employee to branch (QuirkyCroissant)



Use Case 3: Issue Equipment to Agent (LOG1CBOMB)



Use case 4: Promoting an employee (QuirkyCroissant)



Reports

REPORT 1 (LOG1CBOMB):

Report - Find Agents who completed the most successful missions in the last year (365 days) on unique subjects

This report wants to find out which agent has completed the most successful missions in the last year (365 days) on unique subjects. With this information, the bureau can make a educated choice which agent is assigned to a future high-value mission to ensure the highest probability of success possible. Therefore, the report shows a list of agents ranked by their number of successful, completed missions on unique subjects (count_successful_missions_unique_subjects).

Entities: Agent, Missionlog, Subject
Filtered by: Missionlog.date, Missionlog.status
Sorted by: COUNT of successful investigations on unique subjects
Report Columns:
agent.a_id, agent_role, count_successful_missions_unique_subjects, count_unique_subjects, count_successful_missions

REPORT 2 (QuirkyCroissant):

Report - List of history of specific employees past assigned branches and possible transfers

A report that gives detailed information about current and past assignments of an individual employee of the agency to specific associated branches. For improved readability certain attributes taken from the branch entity will get resolved by the Location and Facility_type entities. The report is sorted by the entrance date to a facility(assigned_to.since).

Entities: Employee, Location, Branch, Facility_Type
Filtered by: employee.e_id
Sorted by: assigned_to.since
Report Columns: Branch.b_id, Branch.name, Branch.fk_type, Branch.fk_post_code, assigned_to.since, assigned_to.till

Work Protocol

Author	Date	Task	Result
LOG1CBOMB & QuirkyCroissant	25.03	Finding and agreeing on a reality domain	Espionage Agency as Reality Domain
QuirkyCroissant	27.03	Modelling a ER-Diagram for the chosen reality domain	ERD
LOG1CBOMB	28.03	Refactoring the ER-Diagram	Improved ERD
LOG1CBOMB	29.03	Writing the Business Model Outline with detailed description of the domain and ERD	Business Model Outline
QuirkyCroissant	03.04	Defining Main Use Case 2 and Use Case 4 through description and modelling with BPMN	Use Case 2 and 4
QuirkyCroissant	03.04	Defining the Report for Main Use Case 2	Report
LOG1CBOMB	05.04	Defining Main Use Case 1 and Use Case 3 through description and modelling with BPMN	Use Case 1 and 3
LOG1CBOMB	06.04	Defining the Report for Main Use Case 1	Report
LOG1CBOMB & QuirkyCroissant	11.04	Refactoring Milestone Document for mistakes and final submission	Milestone 1