King Saud University

College of Computer and
Information Sciences

Project CSC343

Management Application for a

Construction Company

F23_08

Suliman AlAkeel

Abdullah AlQahtani

Abdulrahman AlMyman

F23_08

Introduction

We are working on a management application for a construction company using the skills we learned this semester, such as the functional and non-functional, use case diagram, actor-goal list, class diagram, and activity diagram. The goal of the project is to use each of these technologies on the notion.

Table of Contents

F23_08

Functional Requirements:

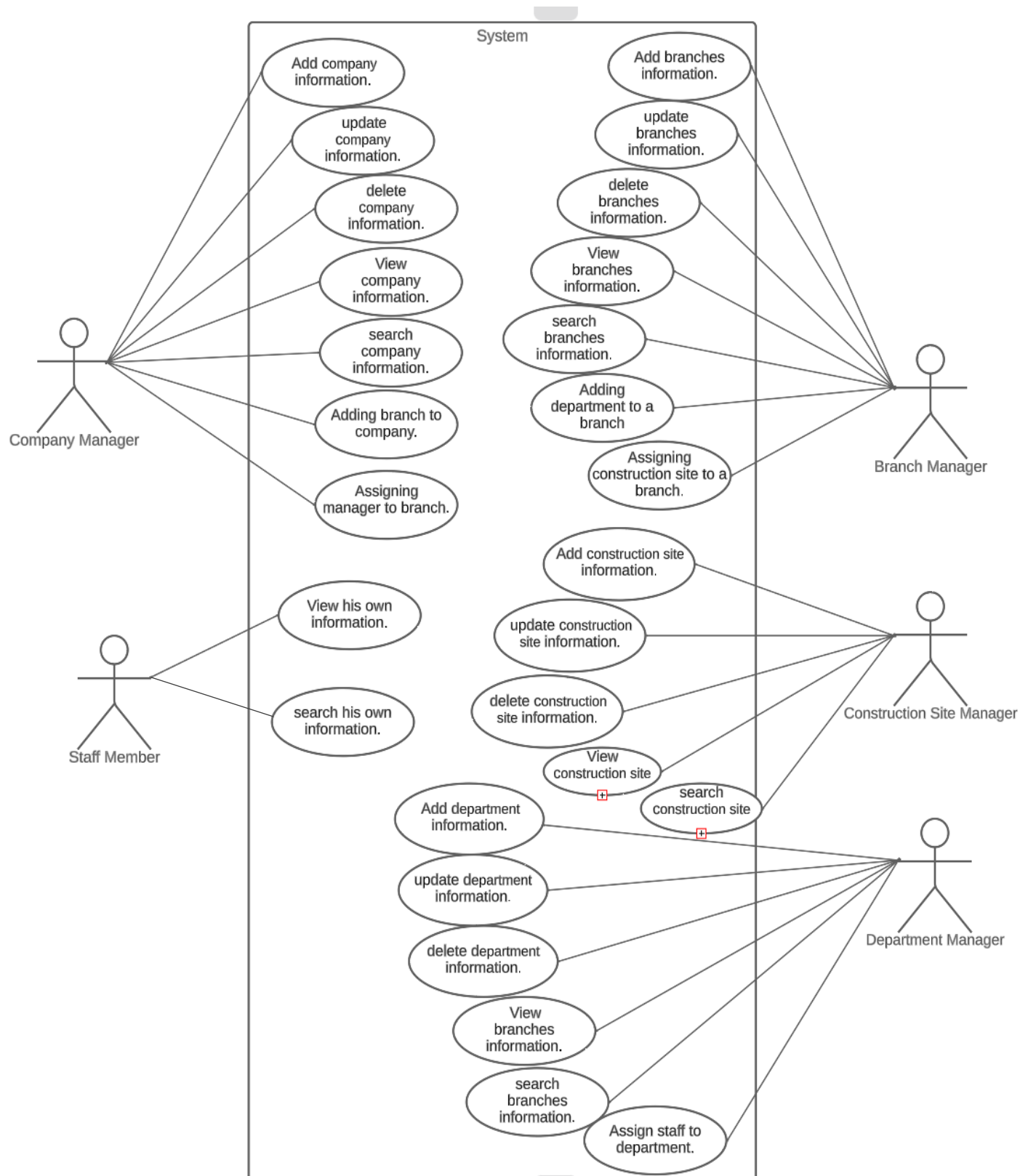| |
|---|
| 1. The application should add company information. |
| 2. The application should update company information. |
| 3. The application should update company information. |
| 4. The application should view company information. |
| 5. The application should search company information. |
| 6. The application should add branches information. |
| 7. The application should update branches information. |
| 8. The application should delete branches information. |
| 9. The application should view branches information. |
| 10. The application should search branches information. |
| 11. The application should add managers information. |
| 12. The application should update managers information. |
| 13. The application should delete managers information. |
| 14. The application should view manages information. |
| 15. The application should search manages information. |
| 16. The application should add department information. |
| 17. The application should update department information. |
| 18. The application should delete department information. |
| 19. The application should view department information. |
| 20. The application should search department information. |
| 21. The application should add staff information. |
| 22. The application should update staff information. |
| 23. The application should delete staff information. |
| 24. The application should view staff information. |
| 25. The application should search staff information. |
| 26. The application should add construction site information. |
| 27. The application should update construction site information. |
| 28. The application should delete construction site information. |
| 29. The application should view construction site information. |
| 30. The application should search construction site information. |
| 31. The application should assign staff to department. |
| 32. The application should adding department to a branch. |
| 33. The application should adding branch to company. |
| 34. The application should assigning manager to branch. |
| 35. The application should assigning construction site to a branch. |

F23_08

Non-Functional Requirements:

1. The Application must be implemented on windows 11.
2. The Application must have a user-friendly Gui.

Actor-Goal Lis:

Actor	Goal
Company Manager	1. Add company information. 2. Update company information. 3. Delete company information. 4. View company information. 5. Search company information. 6. Adding branch to company. 7. Assigning manager to branch.
Branch Manager	1. Add branches information. 2. Update branches information. 3. Delete branches information. 4. View branches information. 5. Search branches information. 6. Adding department to a branch 7. Assigning construction site to a branch.
Department Manager	1. Add department information. 2. Update department information. 3. Delete department information. 4. View department information. 5. Search department information. 6. Assign staff to department.
Staff Member	1. View his own information. 2. Search his own information.
Construction Site Manager	1. Add construction site information. 2. Update construction site information. 3. Delete construction site information. 4. View construction site information. 5. Search construction site information.

F23_08

Use-Case Diagram:

F23_08

Description Of Use-Case:

Situation: The company has many branches, and every branch has a manager. One branch can work for different construction sites. Different branches can work on the same construction site. Every Branch will have the same kind of departments and department will have staff members working for that department.

Use case Name: Adding Staff to Department

Actor: Company Manager

Precondition: Staff has no department.

Post condition: staff are added to Department.

User Actions	**System Response**
i. The Company Manager enters his credentials to the system.	i. System responses with the validation message. **Exp1**
ii. The Company Manager checks if the staff and department's information are correct and available.	ii. System responses with the availability of staff and department message. **Exp2**
iii. The Company Manager Adds the staff to department. **Alt1**	iii. The system responses with the confirmation message.

Exception 1: credentials are not validated. No Adding Staff to Department. Use case ends.
Exception 2: staff or department are not available. Go to step ii or end the use case.
Alternative 1: User directs the system to cancel the operation. Use case ends.

F23_08

<u>***Use case Name: Assigning Construction Site to a branch.***</u>

Actor: Company Manager

Precondition: Construction site is not assigned to a branch.

Post condition: Construction site is assigned to a branch.

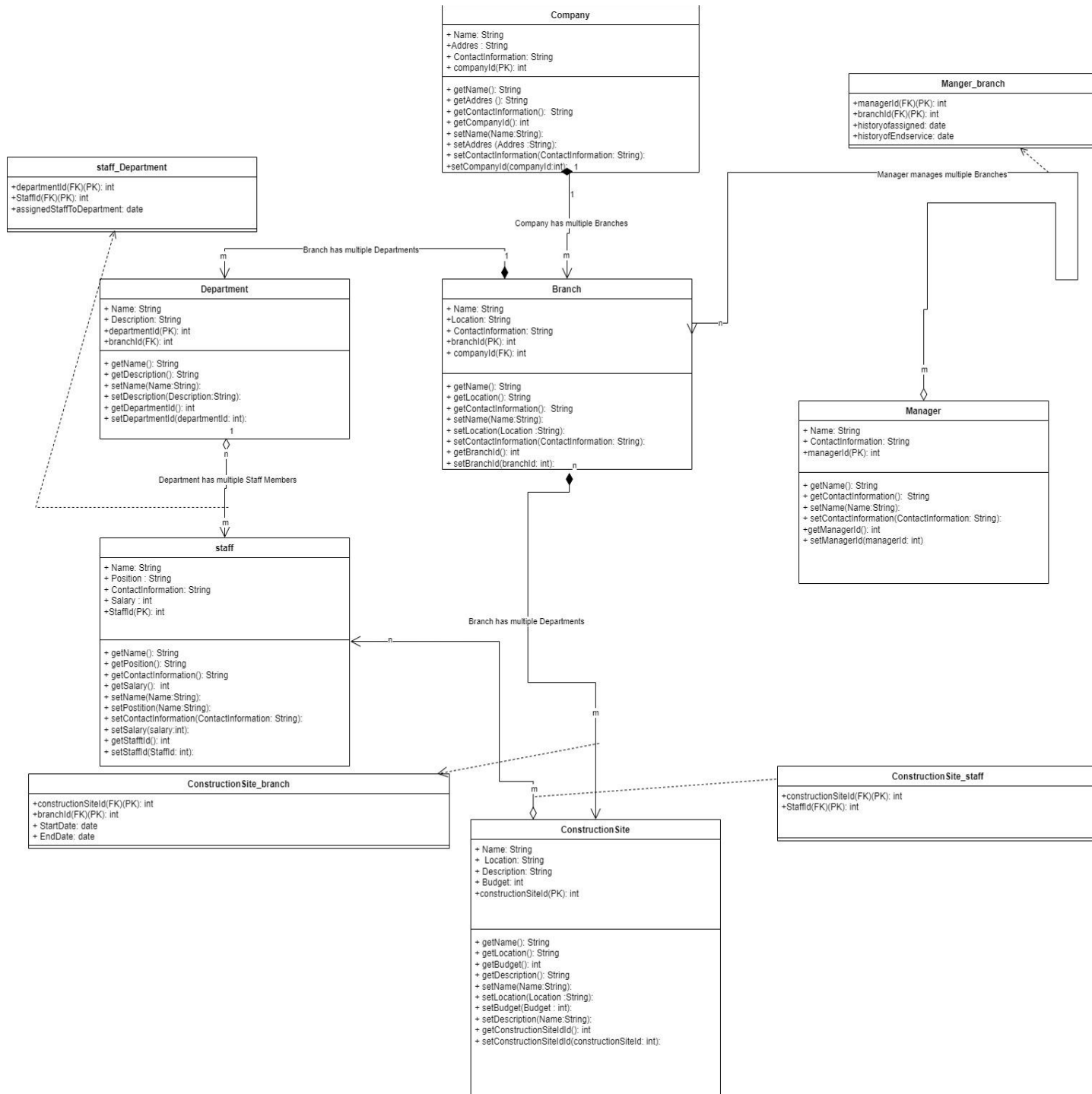| <u>**User Actions**</u>	<u>**System Response**</u>
i. The Company Manager enters his credentials to the system.	i. System responses with the validation message. **Exp1**
ii. The Company Manager checks if the staff and department's information are correct and available.	ii. System responds with the availability of construction site and branch message. **Exp2**
iii. The Company Manager Assign Construction site to department **Alt1**	iii. The system responses with the confirmation message

Exception 1: credentials are not validated. No Assigning Construction Site to a branch. Use case ends.
Exception 2: Construction Site or branch are not available. Go to step ii or end the use case.
Alternative 1: User directs the system to cancel the operation. Use case ends.

F23_08

Class Diagram:

Company

+ Name: String
+Addres : String
+ ContactInformation: String
+ companyId(PK): int

+ getName(): String
+ getAddres (): String
+ getContactInformation(): String
+ getCompanyId(): int
+ setName(Name:String):
+ setAddres (Addres :String):
+ setContactInformation(ContactInformation: String):
+setCompanyId(companyId:int):

Manger_branch

+managerId(FK)(PK): int
+branchId(FK)(PK): int
+historyofassigned: date
+historyofEndservice: date

staff_Department

+departmentId(FK)(PK): int
+StaffId(FK)(PK): int
+assignedStaffToDepartment: date

Company has multiple Branches

Manager manages multiple Branches

Branch has multiple Departments

Department

+ Name: String
+ Description: String
+departmentId(PK): int
+branchId(FK): int

+ getName(): String
+ getDescription(): String
+ setName(Name:String):
+ setDescription(Description:String):
+ getDepartmentId(): int
+ setDepartmentId(departmentId: int):

Branch

+ Name: String
+Location: String
+ ContactInformation: String
+branchId(PK): int
+ companyId(FK): int

+ getName(): String
+ getLocation(): String
+ getContactInformation(): String
+ setName(Name:String):
+ setLocation(Location :String):
+ setContactInformation(ContactInformation: String):
+ getBranchId(): int
+ setBranchId(branchId: int):

Manager

+ Name: String
+ ContactInformation: String
+managerId(PK): int

+ getName(): String
+ getContactInformation(): String
+ setName(Name:String):
+ setContactInformation(ContactInformation: String):
+getManagerId(): int
+ setManagerId(managerId: int)

Department has multiple Staff Members

Branch has multiple Departments

staff

+ Name: String
+ Position : String
+ ContactInformation: String
+ Salary : int
+StaffId(PK): int

+ getName(): String
+ getPosition(): String
+ getContactInformation(): String
+ getSalary(): int
+ setName(Name:String):
+ setPostition(Name:String):
+ setContactInformation(ContactInformation: String):
+ setSalary(salary:int):
+ getStaffId(): int
+ setStaffId(StaffId: int):

ConstructionSite_branch

+constructionSiteId(FK)(PK): int
+branchId(FK)(PK): int
+ StartDate: date
+ EndDate: date

ConstructionSite_staff

+constructionSiteId(FK)(PK): int
+StaffId(FK)(PK): int

ConstructionSite

+ Name: String
+ Location: String
+ Description: String
+ Budget: int
+constructionSiteId(PK): int

+ getName(): String
+ getLocation(): String
+ getBudget(): int
+ getDescription(): String
+ setName(Name:String):
+ setLocation(Location :String):
+ setBudget(Budget : int):
+ setDescription(Name:String):
+ getConstructionSiteIdId(): int
+ setConstructionSiteIdId(constructionSiteId: int):

F23_08

Activity Diagram:

Adding staff to department.

User | System

- User initiates "Adding Staff to Department"
- System prompts for credentials
- User enters credentials
- System validates the credentials
- Credentials are Valid? (yes / no)
- System notifies User of invalid credentials
- System checks the availability of staff and department
- Staff and Department are Available? (yes / no)
- System notifies User that Staff or Department is not available
- System displays a list of available staff and departments
- User checks and confirms staff and department information
- System adds the selected staff to the selected department
- System notifies User of successful addition
- User completes "Adding Staff to Department"