National Autonomous University of Mexico

School of Engineering

Object-oriented programming models

Software requirements

Version :1.0

3 of november 2021

Software requirements

Content:

Software requirements

History of changes

Version	Description	Responsible for the update	Update date
v1.0	document creation	José Alejandro Morán Duque	30/10/2021

Software requirements

1. General description

1.1. Software objectives

The company requires a software that allows the manager to store the name, age, address, worker id, seniority, past projects and current projects of its workers.

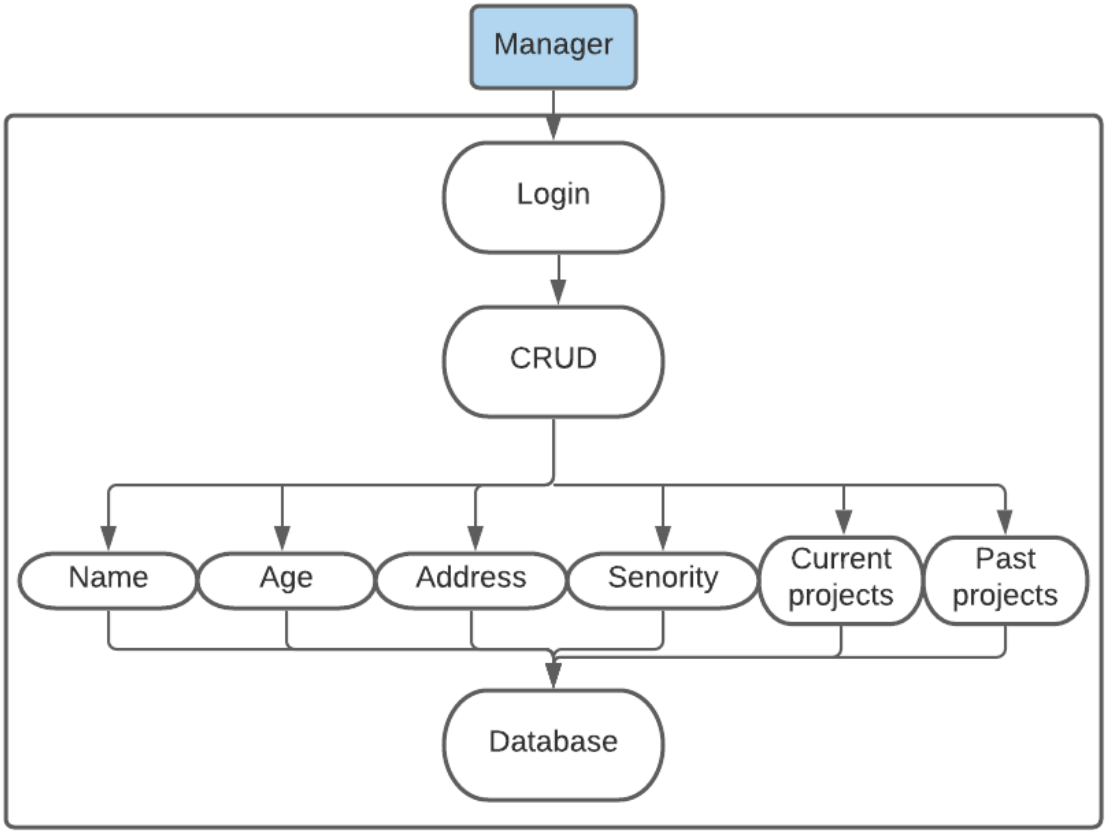
Through the software, the manager will be able to create, read, update and delete the data of any worker**.**

1.2. Glossary of terms

Term	Meaning on the software
Worker	Person that works in the company
Manager	Person that can create, read, update or delete any information in the database
Current projects	Project/s in which the worker is currently working
Past project	Project/s which the worker was a part
Database	.csv document that contains the records of all the workers
Records	Name, age, address, seniority, current and past projects of a worker
Read records	Process in which the manager read on or more records from the database
Write record	Process in which the manager adds a record to the database
Update record	Process in which the manager updates one record in the database
Delete record	Process in which the manager deletes one record of the database
CRUD	Create, Read, Update and Delete process

2. Functional requirements

2.1. Identification of use cases



2.2. Detailed of use cases

Use case	**Login**
Use case description	The manager login to access the functionality of the system
Actors	Manager
Pre-condition	The data gathering must be over
Post-condition	The manager access the system and the CRUD menu is displayed
Basic flow	**Step**
Actor enter the password	1
Controller validates the password	2
Controller allows actor to the system	3
Alternative flow	**Step**
Invalid password Controller shows an error message	2a

Use case	**Create record**
Use case description	The manager create a new record
Actors	Manager
Pre-condition	Manager must be logged
Post-condition	The new record is written in the database

Software requirements

Basic flow	**Step**
Controller show the create view	1
Actor enter the name, age, address, id, seniority, current projects and past projects	2
Controller validates if the id is already in the database	3
Controller write the new record on the database	4
Alternative flow	**Step**
Invalid name Controller shows an error message	2a
Invalid age Controller shows an error message	2b
Invalid address Controller shows an error message	2c
Invalid id Controller shows an error message	2d
Invalid seniority Controller shows an error message	2e
Invalid current projects Controller shows an error message	2f
Invalid past projects Controller shows an error message	2g
The id is on the database Controller shows an error message	3a

Software requirements

Use case	**Read record**
Use case description	The manager read one record
Actors	Manager
Pre-condition	Manager must be logged
Post-condition	The requested record is displayed on screen
Basic flow	**Step**
Actor enter the id of the record	1
Controller search the id on the database	2
Controller display the requested record	3
Alternative flow	**Step**
The id is not on the database Controller shows an error message	2a

Use case	**Update record**
Use case description	The manager read one record
Actors	Manager
Pre-condition	Manager must be logged
Post-condition	The requested record is displayed on screen

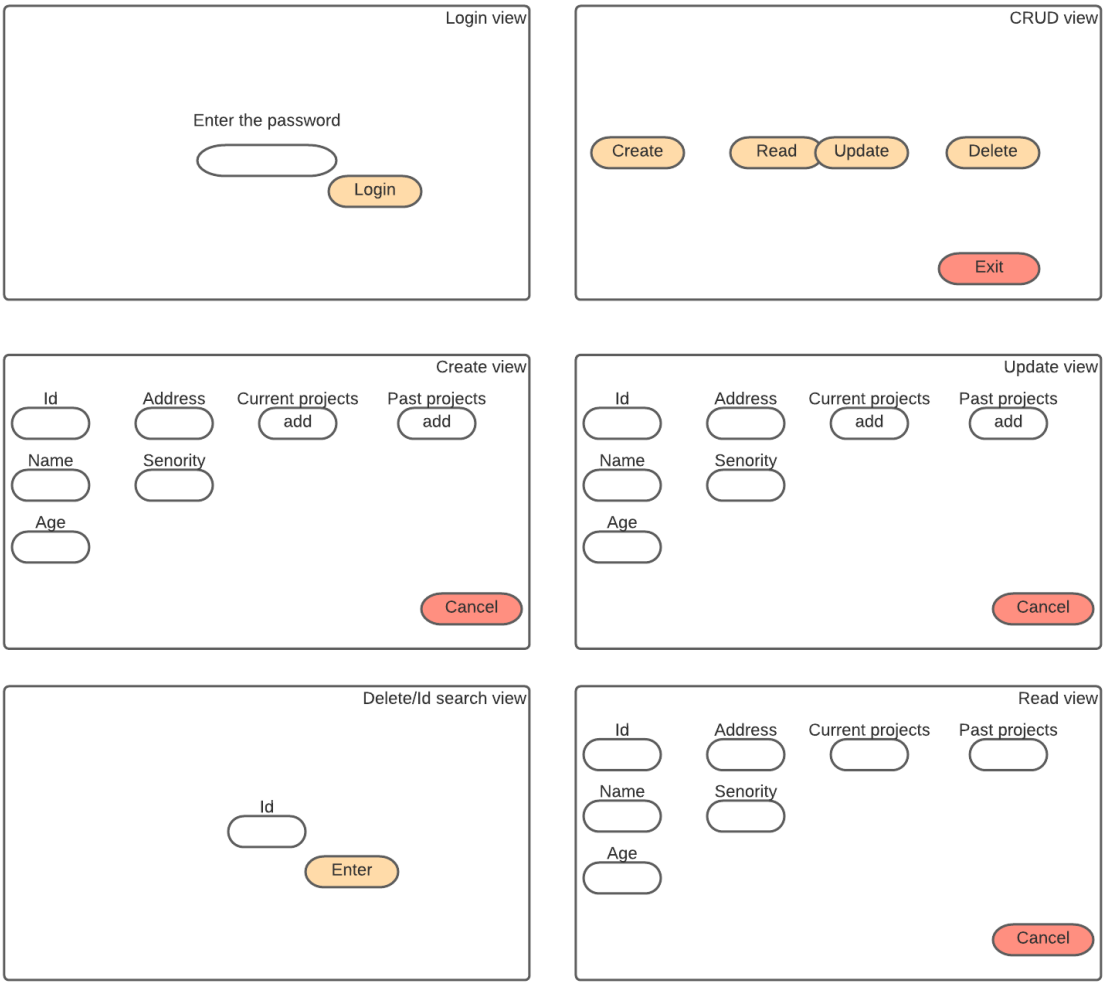
Software requirements

Basic flow	**Step**
Actor enter the id of the record	2
Controller search the id on the database	3
Controller display the requested record on the create view	4
Actor modify zero or more of the: name, age, address, id, seniority, current project or past projects	5
Controller write the modified record on the database	6
Alternative flow	**Step**
The id is not on the database Controller shows an error message	3a
Invalid name Controller shows an error message	5a
Invalid age Controller shows an error message	5b
Invalid address Controller shows an error message	5c
Invalid seniority Controller shows an error message	5d
Invalid past projects Controller shows an error message	5e
Invalid current projects Controller shows an error message	5f

Software requirements

Use case	**Delete record**
Use case description	The manager delete one record
Actors	Manager
Pre-condition	Manager must be logged
Post-condition	The requested record is deleted of the database
Basic flow	**Step**
Actor enter the id of the record	1
Controller search the id on the database	2
Controller delete the requested record	3
Alternative flow	**Step**
The id is not on the database Controller shows an error message	2a

2.3. User interface prototype



2.4. System test plan

The system will be tested by adding 100 records to the database with random generated data obtained from 2 csv files.
When all the records are written to the database we will read all,update one, delete one and then read all again.

3. Functional requirements

3.1. Usability

The ISO/IEC 9126-1:2001 will be followed in order to ensure that the program is easy to use, easy to learn, attractive for the user and conforming with standards/guidelines.

3.2. Reliability

The system must be capable of operating at least 5 days a week and also must have at least one backup of the database from the last month.

3.3. Efficiency

All actions must be read, processed and stored within a maximum of 10 seconds.

3.4. Maintenance

Once the system is deployed every maintenance ticket will have an additional cost.

3.5. Portability

The system will only be able on a Pc/Laptop

3.6. Design and implementation restrictions

The system must be program in java

3.7. Reusability

The system won't buy or use any component from other system

3.8. Interfaces

The system has no interaction with interfaces of other systems, hardware or software.