Solution4

```
EMPLOYEE(enumber, first-name, last-name, project-title,
                              budget, deadline, software-used)
```

According to the relational schema, we can create a table to verify the quality of the relational schema's regulation:

enumber	first-name	last-name	project-title	budget	deadline	software-used
001	John	Bush	CSIT882	1000	Oct.21.2023	StarUML
002	Tom	George	CSIT882	1000	Oct.21.2023	StarUML
003	Amy	King	CSIT882	1000	Oct.21.2023	StarUML

Now that we have drawn a sample form based on the requirements of the relational schema, there is a lot of redundant information generated because all these employees are attached to the same ***project-title*** and have the same project ***budget***, ***deadline***, and ***software-used***.

Steps1:

Separate `EMPLOYEE` and `PROJECT`:

```
EMPLOYEE(enumber, first-name, last-name, project-title)
PRIMARY KEY = (enumber)

PROJECT(project-title, budget, deadline, software-used)
PRIMARY KEY = (project-title)
```

enumber	first-name	last-name	project-title	budget	deadline	software-used
001	John	Bush	CSIT882	1000	Oct.21.2023	StarUML
002	Tom	George	CSIT882	1000	Oct.21.2023	StarUML
003	Amy	King	CSIT882	1000	Oct.21.2023	StarUML
004	Mike	Card	CSIT883	2000	Oct.29.2023	MS_OFFICE
005	John	Dan	CSIT883	2000	Oct.29.2023	WPS_OFFICE

Steps2:

Now we have inserted two entities, both belonging to the same project, but using different software. In this instance, to handle the **many-to-many** relationship between `PROJECT` and `SOFTWARE`, I used a linking table called `PROJECT_SOFTWARE`.

```
EMPLOYEE(enumber, first-name, last-name, project-title)
PRIMARY KEY = (enumber)
FOREIGN KEY = (project-title) REFERENCES PROJECT(project-
title)
```

```
PROJECT(project-title, budget, deadline)
PRIMARY KEY = (project-title)

SOFTWARE(software-used)
PRIMARY KEY = (software-used)

PROJECT_SOFTWARE(project-title, software-used)
PRIMARY KEY = (project-title, software-used)
FOREIGN KEY = (project-title) REFERENCES PROJECT(project-
title)
FOREIGN KEY = (software-used) REFERENCES SOFTWARE(software-
used)
```

Now we add more entities to verify the relational schema:

enumber	first-name	last-name	project-title	budget	deadline	software-used
001	John	Bush	CSIT882	1000	Oct.21.2023	StarUML
002	Tom	George	CSIT882	1000	Oct.21.2023	StarUML
003	Amy	King	CSIT882	1000	Oct.21.2023	StarUML
004	Mike	Card	CSIT883	2000	Oct.29.2023	MS_OFFICE
005	John	Dan	CSIT883	2000	Oct.29.2023	WPS_OFFICE
006	Oliver	Che	CSIT882	1000	Oct.21.2023	UML_TOOLS
007	George	Town	CSIT883	2000	Oct.29.2023	LaTeX

After verification, the relational schema conforms to the requirements and relationships described in the original description.