

## **A5 – Relational schema, validation and schema refinement**

This artifact contains the relational schema obtained by mapping the conceptual data model made in the previous week, which allows a different approach of the entities and relationship involved in a project management web application.

### **1. Relational Schema**

The relational schema includes the relational schema attributes, domains, keys (primary and foreign) and other integrity rules.

<b>R01</b>	User ( <u>id</u> , e-mail <b>NN UK</b> , full_name, password <b>NN</b> , URL <b>NN UK</b> , username <b>NN UK</b> , administrator <b>NN DF 0</b> )
<b>R02</b>	Project ( <u>id</u> , description, end_date <b>CK</b> end_date > start_date, name <b>NN</b> , privacy <b>DF</b> privacy=0, start_date <b>NN DF</b> today, state <b>NN CK</b> state <b>IN</b> stateType <b>DF</b> 'In_progress', id_coordinator → User)
<b>R03</b>	Board ( <u>id</u> , description, name <b>NN</b> , id_creator → User, id_project → Project)
<b>R04</b>	Task ( <u>id</u> , budget, deadline <b>NN</b> , description, name <b>NN</b> , progress <b>CK</b> 0 ≤ progress ≤ 100, state <b>NN CK</b> state <b>IN</b> stateType <b>DF</b> 'In_progress', id_creator → User, id_board → Board)
<b>R05</b>	Meeting ( <u>id</u> , date <b>NN CK</b> date > today, name, place, id_board → Board)
<b>R06</b>	Personal_event ( <u>id</u> , date <b>NN CK</b> date > today, name, place, id_user → User)
<b>R07</b>	Comment ( <u>id</u> , comment <b>NN</b> , date <b>NN DF</b> today, id_user → User, id_task → Task)
<b>R08</b>	File ( <u>id</u> , description, path <b>NN</b> , publish_date <b>NN DF</b> today, id_user → User, id_task → Task)
<b>R09</b>	Notification ( <u>id</u> , date <b>NN DF</b> today, notification <b>NN</b> , read <b>NN DF 0</b> , id_user → User)
<b>R10</b>	Progress_update ( <u>id</u> , date <b>NN DF</b> today, new_value <b>CK</b> 0 ≤ new_value ≤ 100, id_user → User, id_task → Task)
<b>R11</b>	Message ( <u>id</u> , date <b>NN DF</b> today, message <b>NN</b> , id_user → User, id_project → Project)
<b>R12</b>	Project_team ( <u>id_project</u> → Project, <u>id_worker</u> → User)
<b>R13</b>	Board_team ( <u>id_board</u> → Board, <u>id_worker</u> → User)
<b>R14</b>	Profile_picture ( <u>id_image</u> , id_user → User, path <b>UK</b> )
<b>R15</b>	Project_picture ( <u>id_image</u> , id_project → Project, path <b>UK</b> )
<b>R16</b>	Contact ( <u>id_user</u> → User, <u>id_contact</u> → User)
<b>R17</b>	Ban_user ( <u>id_administrator</u> → User, <u>id_user</u> → User, date)
<b>R18</b>	Archive_project ( <u>id_project</u> → Project, <u>id_administrator</u> → User, date)

Relational schemas are presented above, where UK means UNIQUE KEY, NN means NOT NULL, DF means DEFAULT and CK means CHECK.

### **2. Domains**

Specification of additional domains:

today	DATE DEFAULT CURRENT_DATE
stateType	ENUM ('Closed', 'In_progress', 'Archived')

### 3. Functional dependencies and schema validation

To validate the Relational Schema obtained from the Conceptual Model, all functional dependencies are identified, and the normalization of all relation schemas is accomplished. All relations schemas are in the Boyce-Codd Normal Form, so the relational schema is also in the BCNF and there is no need to be refined using normalization.

<b>Table R01 (User)</b>	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0101	{id} → {e-mail, full_name, password, URL, username, administrator}
<b>NORMAL FORM</b>	BCNF

<b>Table R02 (Project)</b>	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0201	{id} → {description, end_date, name, privacy, start_date, state, id_coordinator}
<b>NORMAL FORM</b>	BCNF

<b>Table R03 (Board)</b>	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0301	{id} → {description, name, id_creator, id_project}
<b>NORMAL FORM</b>	BCNF

<b>Table R04 (Task)</b>	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0401	{id} → {budget, deadline, description, name, progress, state, id_creator, id_board}
<b>NORMAL FORM</b>	BCNF

<b>Table R05 (Meeting)</b>	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0501	{id} → {date, name, place, id_board}
<b>NORMAL FORM</b>	BCNF

<b>Table R06 (Personal_event)</b>	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0601	{id} → {date, name, place, id_user}
<b>NORMAL FORM</b>	BCNF

<b>Table R07</b> (Comment)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0701	{id} → {comment, date, id_user, id_task}
<b>NORMAL FORM</b>	BCNF

<b>Table R08</b> (File)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD0801	{id} → {description, path, publish_date, id_user, id_task}
<b>NORMAL FORM</b>	BCNF

<b>Table R09</b> (Notification)	
<b>Keys:</b> {id, id_user}	
<b>Functional Dependencies</b>	
FD0901	{id, id_user} → {date, notification, state}
<b>NORMAL FORM</b>	BCNF

<b>Table R10</b> (Progress_update)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD1001	{id} → {date, new_value, id_user, id_task}
<b>NORMAL FORM</b>	BCNF

<b>Table R11</b> (Message)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD1101	{id} → {date, message, id_user, id_project}
<b>NORMAL FORM</b>	BCNF

<b>Table R12</b> (Project_team)	
<b>Keys:</b> {id_project, id_worker}	
<b>Functional Dependencies</b>	
(none)	
<b>NORMAL FORM</b>	BCNF

<b>Table R13</b> (Board_team)	
<b>Keys:</b> {id_board, id_worker}	
<b>Functional Dependencies</b>	
(none)	
<b>NORMAL FORM</b>	BCNF

<b>Table R14</b> (Profile_picture)	
<b>Keys:</b> {id_image}	
<b>Functional Dependencies</b>	
FD1401	id_image → {id_user, path}
<b>NORMAL FORM</b>	BCNF

<b>Table R15</b> (Project_picture)	
<b>Keys:</b> {id_image}	
<b>Functional Dependencies</b>	
FD1501	id_image → {id_project, path}
<b>NORMAL FORM</b>	BCNF

<b>Table R16</b> (Contact)	
<b>Keys:</b> {id_user, id_contact}	
<b>Functional Dependencies</b>	
(none)	
<b>NORMAL FORM</b>	BCNF

<b>Table R17</b> (Ban_user)	
<b>Keys:</b> {id_user}	
<b>Functional Dependencies</b>	
FD1701	id_user → {id_administrator}
<b>NORMAL FORM</b>	BCNF

<b>Table R18</b> (Archive_project)	
<b>Keys:</b> {id_project}	
<b>Functional Dependencies</b>	
FD1801	id_project → {id_administrator}
<b>NORMAL FORM</b>	BCNF

#### 4. SQL Code

(<https://github.com/TeresaValerio/LBAW1734/blob/master/Artefactos/A5/database.sql>)

```
--
-- Tables
--
CREATE TABLE "Archive_project" (
    id_administrator integer NOT NULL,
    id_project integer NOT NULL,
    date timestamp without time zone DEFAULT now() NOT NULL
);
```

```
CREATE TABLE "Ban_user" (  
    id_user integer NOT NULL,  
    id_administrator integer NOT NULL,  
    date timestamp without time zone DEFAULT now() NOT NULL  
);
```

```
CREATE TABLE "Board" (  
    id integer NOT NULL,  
    description text,  
    name text NOT NULL,  
    id_creator integer NOT NULL,  
    id_project integer NOT NULL  
);
```

```
CREATE TABLE "Board_team" (  
    id_board integer NOT NULL,  
    id_user integer NOT NULL  
);
```

```
CREATE TABLE "Comment" (  
    id integer NOT NULL,  
    comment text NOT NULL,  
    date timestamp without time zone DEFAULT now() NOT NULL,  
    id_user integer NOT NULL,  
    id_task integer NOT NULL  
);
```

```
CREATE TABLE "Contact" (  
    id_user integer NOT NULL,  
    id_contact integer NOT NULL  
);
```

```
CREATE TABLE "File" (  
    id integer NOT NULL,  
    path text NOT NULL,  
    publish_date timestamp without time zone DEFAULT now() NOT NULL,  
    description text,  
    id_user integer NOT NULL,  
    id_task integer NOT NULL  
);
```

```

CREATE TABLE "Meeting" (
    id integer NOT NULL,
    date timestamp without time zone NOT NULL,
    name text,
    place text,
    id_board integer NOT NULL,
    CONSTRAINT "CK1" CHECK ((date > now()))
);

CREATE TABLE "Message" (
    id integer NOT NULL,
    message text NOT NULL,
    date timestamp without time zone DEFAULT now() NOT NULL,
    id_user integer NOT NULL,
    id_project integer NOT NULL
);

CREATE TABLE "Notification" (
    id integer NOT NULL,
    id_user integer NOT NULL,
    date timestamp without time zone DEFAULT now() NOT NULL,
    notification text NOT NULL,
    read boolean DEFAULT false NOT NULL
);

CREATE TABLE "Personal_event" (
    id integer NOT NULL,
    date timestamp without time zone NOT NULL,
    place text,
    name text,
    id_user integer NOT NULL,
    CONSTRAINT "CK1" CHECK ((date > now()))
);

CREATE TABLE "Profile_picture" (
    id integer NOT NULL,
    id_user integer NOT NULL,
    path text NOT NULL
);

```

```
CREATE TABLE "Progress_update" (  
    id integer NOT NULL,  
    date timestamp without time zone DEFAULT now() NOT NULL,  
    new_value integer NOT NULL,  
    id_user integer NOT NULL,  
    id_task integer NOT NULL,  
    CONSTRAINT "CK1" CHECK ((new_value > 0)),  
    CONSTRAINT "CK2" CHECK ((new_value <= 100))  
);
```

```
CREATE TABLE "Project" (  
    id integer NOT NULL,  
    description text,  
    start_date timestamp without time zone DEFAULT now() NOT NULL,  
    end_date date,  
    name text NOT NULL,  
    id_coordinator integer NOT NULL,  
    state text DEFAULT 'In_progress' NOT NULL,  
    privacy boolean DEFAULT false NOT NULL,  
    CONSTRAINT "CK1" CHECK ((end_date > start_date)),  
    CONSTRAINT "CK2" CHECK ((state = ANY (ARRAY['In_progress'::text,  
'Closed'::text, 'Archived'::text])))  
);
```

```
CREATE TABLE "Project_picture" (  
    id integer NOT NULL,  
    id_project integer NOT NULL,  
    path text NOT NULL  
);
```

```
CREATE TABLE "Project_team" (  
    id_user integer NOT NULL,  
    id_project integer NOT NULL  
);
```

```
CREATE TABLE "Task" (  
    id integer NOT NULL,  
    budget money,  
    deadline date NOT NULL,  
    description text,  
    name text NOT NULL,
```

```

progress integer DEFAULT 0 NOT NULL,
state text DEFAULT 'In_progress' NOT NULL,
id_creator integer NOT NULL,
id_board integer NOT NULL,
CONSTRAINT "CK1" CHECK ((progress > 0)),
CONSTRAINT "CK2" CHECK ((progress <= 100)),
CONSTRAINT "CK3" CHECK ((state = ANY (ARRAY['In_progress'::text,
'Closed'::text, 'Archived'::text])))
);

CREATE TABLE "User" (
    id integer NOT NULL,
    "e-mail" text NOT NULL,
    full_name text,
    password text NOT NULL,
    "URL" text NOT NULL,
    username text NOT NULL,
    administrator boolean DEFAULT false NOT NULL
);

--
-- Primary Keys and Uniques
--
ALTER TABLE ONLY "Archive_project"
    ADD CONSTRAINT "Archive_project_pkey" PRIMARY KEY (id_administrator);

ALTER TABLE ONLY "Ban_user"
    ADD CONSTRAINT "Ban_user_pkey" PRIMARY KEY (id_user);

ALTER TABLE ONLY "Board"
    ADD CONSTRAINT "Board_pkey" PRIMARY KEY (id);

ALTER TABLE ONLY "Board_team"
    ADD CONSTRAINT "Board_team_pkey" PRIMARY KEY (id_board, id_user);

ALTER TABLE ONLY "Comment"
    ADD CONSTRAINT "Comment_pkey" PRIMARY KEY (id);

ALTER TABLE ONLY "Contact"

```



```
ADD CONSTRAINT "Contact_pkey" PRIMARY KEY (id_user, id_contact);
```

```
ALTER TABLE ONLY "File"  
ADD CONSTRAINT "File_path_key" UNIQUE (path);
```

```
ALTER TABLE ONLY "File"  
ADD CONSTRAINT "File_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Meeting"  
ADD CONSTRAINT "Meeting_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Message"  
ADD CONSTRAINT "Message_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Notification"  
ADD CONSTRAINT "Notification_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Personal_event"  
ADD CONSTRAINT "Personal_event_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Profile_picture"  
ADD CONSTRAINT "Profile_picture_path_key" UNIQUE (path);
```

```
ALTER TABLE ONLY "Profile_picture"  
ADD CONSTRAINT "Profile_picture_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Progress_update"  
ADD CONSTRAINT "Progress_update_pkey" PRIMARY KEY (id);
```

```
ALTER TABLE ONLY "Project_picture"  
ADD CONSTRAINT "Project_picture_path_key" UNIQUE (path);
```

```
ALTER TABLE ONLY "Project_picture"  
ADD CONSTRAINT "Project_picture_pkey" PRIMARY KEY (id);
```

```

ALTER TABLE ONLY "Project"
    ADD CONSTRAINT "Project_pkey" PRIMARY KEY (id);

ALTER TABLE ONLY "Project_team"
    ADD CONSTRAINT "Project_team_pkey" PRIMARY KEY (id_user, id_project);

ALTER TABLE ONLY "Task"
    ADD CONSTRAINT "Task_pkey" PRIMARY KEY (id);

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_URL_key" UNIQUE ("URL");

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_e-mail_key" UNIQUE ("e-mail");

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_pkey" PRIMARY KEY (id);

ALTER TABLE ONLY "User"
    ADD CONSTRAINT "User_username_key" UNIQUE (username);

--
-- Foreign Keys
--
ALTER TABLE ONLY "Archive_project"
    ADD CONSTRAINT "Archive_project_id_administrator_fkey" FOREIGN KEY
(id_administrator) REFERENCES "User"(id);

ALTER TABLE ONLY "Archive_project"
    ADD CONSTRAINT "Archive_project_id_project_fkey" FOREIGN KEY
(id_project) REFERENCES "Project"(id);

ALTER TABLE ONLY "Ban_user"
    ADD CONSTRAINT "Ban_user_id_administrator_fkey" FOREIGN KEY
(id_administrator) REFERENCES "User"(id);

```

```
ALTER TABLE ONLY "Ban_user"  
    ADD CONSTRAINT "Ban_user_id_user_fkey" FOREIGN KEY (id_user) REFERENCES  
"User"(id);
```

```
ALTER TABLE ONLY "Board"  
    ADD CONSTRAINT "Board_id_creator_fkey" FOREIGN KEY (id_creator)  
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Board"  
    ADD CONSTRAINT "Board_id_project_fkey" FOREIGN KEY (id_project)  
REFERENCES "Project"(id);
```

```
ALTER TABLE ONLY "Board_team"  
    ADD CONSTRAINT "Board_team_id_board_fkey" FOREIGN KEY (id_board)  
REFERENCES "Board"(id);
```

```
ALTER TABLE ONLY "Board_team"  
    ADD CONSTRAINT "Board_team_id_user_fkey" FOREIGN KEY (id_user)  
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Comment"  
    ADD CONSTRAINT "Comment_id_task_fkey" FOREIGN KEY (id_task) REFERENCES  
"Task"(id);
```

```
ALTER TABLE ONLY "Comment"  
    ADD CONSTRAINT "Comment_id_user_fkey" FOREIGN KEY (id_user) REFERENCES  
"User"(id);
```

```
ALTER TABLE ONLY "Contact"  
    ADD CONSTRAINT "Contact_id_contact_fkey" FOREIGN KEY (id_contact)  
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Contact"  
    ADD CONSTRAINT "Contact_id_user_fkey" FOREIGN KEY (id_user) REFERENCES  
"User"(id);
```

```
ALTER TABLE ONLY "File"  
    ADD CONSTRAINT "File_id_task_fkey" FOREIGN KEY (id_task) REFERENCES  
"Task"(id);
```

```
ALTER TABLE ONLY "File"  
    ADD CONSTRAINT "File_id_user_fkey" FOREIGN KEY (id_user) REFERENCES  
"User"(id);
```

```
ALTER TABLE ONLY "Meeting"  
    ADD CONSTRAINT "Meeting_id_board_fkey" FOREIGN KEY (id_board)  
REFERENCES "Board"(id);
```

```
ALTER TABLE ONLY "Message"  
    ADD CONSTRAINT "Message_id_project_fkey" FOREIGN KEY (id_project)  
REFERENCES "Project"(id);
```

```
ALTER TABLE ONLY "Message"  
    ADD CONSTRAINT "Message_id_user_fkey" FOREIGN KEY (id_user) REFERENCES  
"User"(id);
```

```
ALTER TABLE ONLY "Notification"  
    ADD CONSTRAINT "Notification_id_user_fkey" FOREIGN KEY (id_user)  
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Personal_event"  
    ADD CONSTRAINT "Personal_event_id_user_fkey" FOREIGN KEY (id_user)  
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Profile_picture"  
    ADD CONSTRAINT "Profile_picture_id_user_fkey" FOREIGN KEY (id_user)  
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Progress_update"  
    ADD CONSTRAINT "Progress_update_id_task_fkey" FOREIGN KEY (id_task)  
REFERENCES "Task"(id);
```

```
ALTER TABLE ONLY "Progress_update"
```

```
ADD CONSTRAINT "Progress_update_id_user_fkey" FOREIGN KEY (id_user)
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Project"
ADD CONSTRAINT "Project_id_coordinator_fkey" FOREIGN KEY
(id_coordinator) REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Project_picture"
ADD CONSTRAINT "Project_picture_id_project_fkey" FOREIGN KEY
(id_project) REFERENCES "Project"(id);
```

```
ALTER TABLE ONLY "Project_team"
ADD CONSTRAINT "Project_team_id_project_fkey" FOREIGN KEY (id_project)
REFERENCES "Project"(id);
```

```
ALTER TABLE ONLY "Project_team"
ADD CONSTRAINT "Project_team_id_user_fkey" FOREIGN KEY (id_user)
REFERENCES "User"(id);
```

```
ALTER TABLE ONLY "Task"
ADD CONSTRAINT "Task_id_board_fkey" FOREIGN KEY (id_board) REFERENCES
"Board"(id);
```

```
ALTER TABLE ONLY "Task"
ADD CONSTRAINT "Task_id_creator_fkey" FOREIGN KEY (id_creator)
REFERENCES "User"(id);
```

GROUP1734, 18/03/2018

- Ilona Generalova, up201400035@fe.up.pt
- Maria Inês Gonçalves, up201402784@fe.up.pt
- Maria Teresa Valério, up201405655@fe.up.pt
- Sara Gomes, up201405085@fe.up.pt