

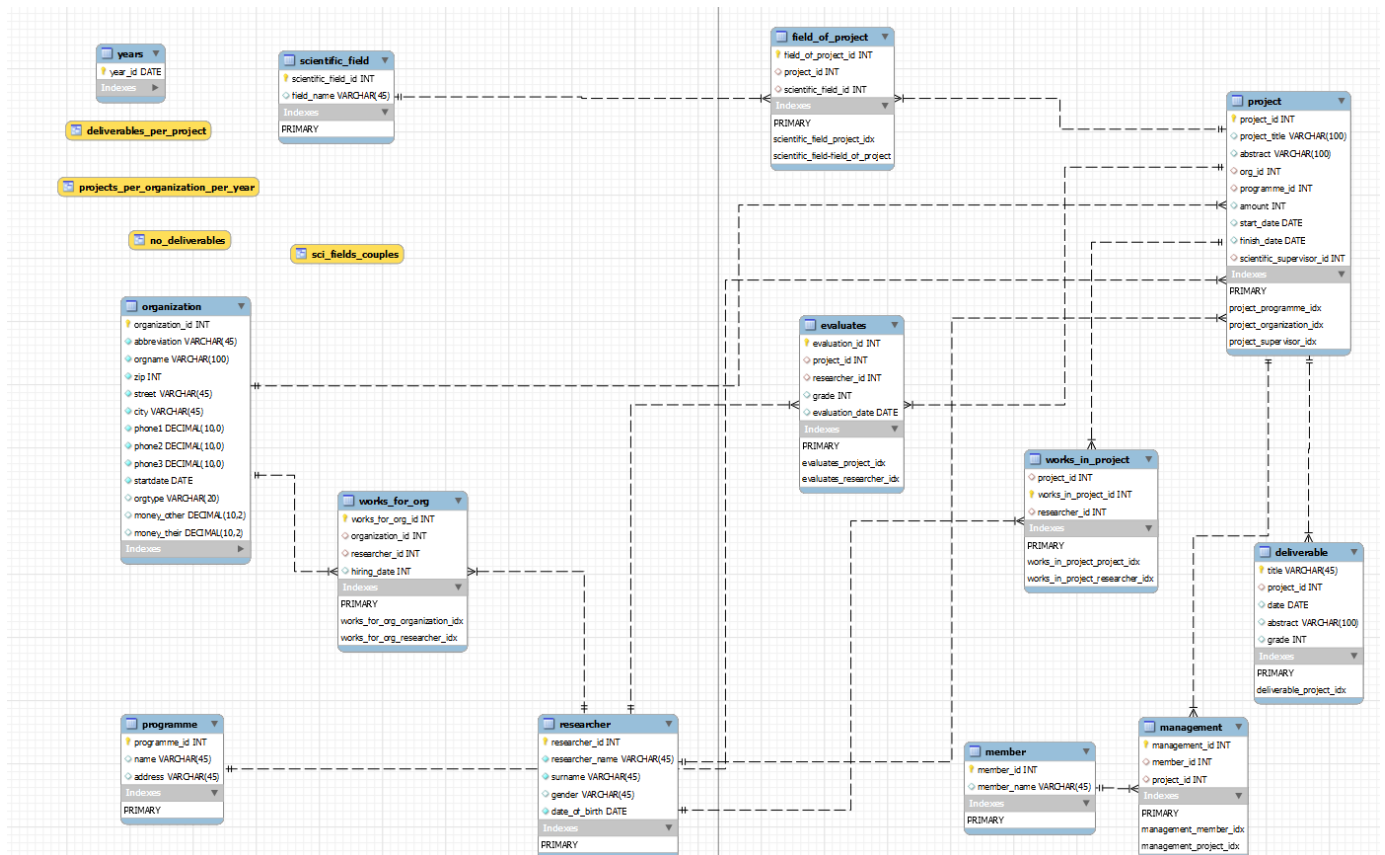
## Βάσεις Δεδομένων

### Αναφορά Εξαμηνιαίας Εργασίας

Νικόλαος Σφακιανάκης (e19130) Ανδρέας Χαραλαμπίδης (e19134)

Νικόλαος Χατζής (e19117)

#### 1. Σχεσιακό Διάγραμμα



1.1 Στην αριστερή πάνω γωνία φαίνονται views που δημιουργήθηκαν για την απάντηση των ερωτημάτων.

1.2 Για τη δημιουργία των παραπάνω πινάκων στην SQL απαιτήσαμε να μην μένουν χωρίς τιμή (NULL ) μερικά βασικά πεδία, όπως primary/foreign keys.

Όπως φαίνεται και στο αρχείο Tables\_2.sql χρησιμοποιήθηκαν οι παρακάτω εντολές για τη δημιουργία της βάσης και των πινάκων :

```
1  --
2  -- Schema Project
```

```

3  -----
4  CREATE SCHEMA IF NOT EXISTS 'Project' ;
5  USE 'Project' ;
6
7  -----
8  -- Table 'Project'.'programme'
9  -----
10 CREATE TABLE IF NOT EXISTS 'Project'.'programme' (
11     'programme_id' INT NOT NULL AUTO_INCREMENT,
12     'name' VARCHAR(45) NULL,
13     'address' VARCHAR(45) NULL,
14     PRIMARY KEY ('programme_id'))
15 ENGINE = InnoDB;
16
17
18 -----
19 -- Table 'Project'.'organization'
20 -----
21 CREATE TABLE IF NOT EXISTS 'Project'.'organization' (
22     'organization_id' INT NOT NULL AUTO_INCREMENT,
23     'abbreviation' VARCHAR(45) NOT NULL,
24     'orgname' VARCHAR(100) NOT NULL,
25     'zip' INT NOT NULL,
26     'street' VARCHAR(45) NOT NULL,
27     'city' VARCHAR(45) NOT NULL,
28     'phone1' NUMERIC(10,0) NOT NULL,
29     'phone2' NUMERIC(10,0) NOT NULL,
30     'phone3' NUMERIC(10,0) NOT NULL,
31     'startdate' DATE NOT NULL,
32     'orgtype' VARCHAR(20) ,
33     'money_other' NUMERIC(10,2) NULL,
34     'money_their' NUMERIC(10,2) NULL,
35     PRIMARY KEY ('organization_id'),
36     CONSTRAINT chk_type CHECK (orgtype IN ('', ' ', ' '),
37 )
38 ENGINE = InnoDB;
39
40
41 -----
42 -- Table 'Project'.'researcher'
43 -----
44 CREATE TABLE IF NOT EXISTS 'Project'.'researcher' (
45     'researcher_id' INT NOT NULL AUTO_INCREMENT,
46     'researcher_name' VARCHAR(45) NOT NULL,
47     'surname' VARCHAR(45) NOT NULL,
48     'gender' VARCHAR(45) NULL,
49     'date_of_birth' DATE NOT NULL,
50     PRIMARY KEY ('researcher_id'))
51 ENGINE = InnoDB;
52
53
54 -----
55 -- Table 'Project'.'project'

```

```

56  -----
57  CREATE TABLE IF NOT EXISTS 'Project'.'project' (
58      'project_id' INT NOT NULL AUTO_INCREMENT,
59      'project_title' VARCHAR(100) NULL,
60      'abstract' VARCHAR(100) NULL,
61      'org_id' INT NULL,
62      'programme_id' INT NULL,
63      'amount' INT NULL,
64      'start_date' DATE NULL,
65      'finish_date' DATE NULL,
66      'scientific_supervisor_id' INT NULL,
67      CONSTRAINT CHK_Duration CHECK (DATEDIFF(finish_date,start_date)>
        364 and DATEDIFF(finish_date,start_date)< 1460),
68      CONSTRAINT CHK_Amount CHECK (amount>=100000 and amount <= 1000000),
69      PRIMARY KEY ('project_id'),
70      INDEX 'project_programme_idx' (('programme_id' ASC) VISIBLE,
71      INDEX 'project_organization_idx' (('org_id' ASC) VISIBLE,
72      INDEX 'project_supervisor_idx' (('scientific_supervisor_id' ASC)
        INVISIBLE,
73      CONSTRAINT 'project_programme'
74      FOREIGN KEY ('programme_id')
75      REFERENCES 'Project'.'programme' (('programme_id')
76      ON DELETE CASCADE
77      ON UPDATE CASCADE,
78      CONSTRAINT 'project_organization'
79      FOREIGN KEY ('org_id')
80      REFERENCES 'Project'.'organization' (('organization_id')
81      ON DELETE CASCADE
82      ON UPDATE CASCADE,
83      CONSTRAINT 'project_supervisor'
84      FOREIGN KEY ('scientific_supervisor_id')
85      REFERENCES 'Project'.'researcher' (('researcher_id')
86      ON DELETE CASCADE
87      ON UPDATE CASCADE)
88  ENGINE = InnoDB;
89
90
91  -----
92  -- Table 'Project'.'member'
93  -----
94  CREATE TABLE IF NOT EXISTS 'Project'.'member' (
95      'member_id' INT NOT NULL AUTO_INCREMENT,
96      'member_name' VARCHAR(45) NULL,
97      PRIMARY KEY ('member_id'))
98  ENGINE = InnoDB;
99
100
101  -----
102  -- Table 'Project'.'deliverable'
103  -----
104  CREATE TABLE IF NOT EXISTS 'Project'.'deliverable' (
105      'title' VARCHAR(45) NOT NULL,
106      'project_id' INT NULL,
107      'date' DATE NULL,

```

```

108 'abstract' VARCHAR(100) NULL,
109 'grade' INT NULL,
110 PRIMARY KEY ('title'),
111 INDEX 'deliverable_project_idx' ('project_id' ASC) VISIBLE,
112 CONSTRAINT 'deliverable_project'
113     FOREIGN KEY ('project_id')
114     REFERENCES 'Project' ('project' ('project_id'))
115     ON DELETE CASCADE
116     ON UPDATE CASCADE)
117 ENGINE = InnoDB;
118
119
120 -----
121 -- Table 'Project'.'scientific_field'
122 -----
123 CREATE TABLE IF NOT EXISTS 'Project'.'scientific_field' (
124     'scientific_field_id' INT NOT NULL AUTO_INCREMENT,
125     'field_name' VARCHAR(45) NULL,
126     PRIMARY KEY ('scientific_field_id'))
127 ENGINE = InnoDB;
128
129
130 -----
131 -- Table 'Project'.'field_of_project'
132 -----
133 CREATE TABLE IF NOT EXISTS 'Project'.'field_of_project' (
134     'field_of_project_id' INT NOT NULL AUTO_INCREMENT,
135     'project_id' INT NULL,
136     'scientific_field_id' INT NULL,
137     PRIMARY KEY ('field_of_project_id'),
138     INDEX 'scientific_field_project_idx' ('project_id' ASC) VISIBLE,
139     CONSTRAINT 'scientific_field-field_of_project'
140     FOREIGN KEY ('scientific_field_id')
141     REFERENCES 'Project'.'scientific_field' ('scientific_field_id')
142     ON DELETE CASCADE
143     ON UPDATE CASCADE,
144     CONSTRAINT 'field_project'
145     FOREIGN KEY ('project_id')
146     REFERENCES 'Project' ('project' ('project_id'))
147     ON DELETE CASCADE
148     ON UPDATE CASCADE)
149 ENGINE = InnoDB;
150
151
152 -----
153 -- Table 'Project'.'management'
154 -----
155 CREATE TABLE IF NOT EXISTS 'Project'.'management' (
156     'management_id' INT NOT NULL AUTO_INCREMENT,
157     'member_id' INT NULL,
158     'project_id' INT NULL,
159     PRIMARY KEY ('management_id'),
160     INDEX 'management_member_idx' ('member_id' ASC) INVISIBLE,
161     INDEX 'management_project_idx' ('project_id' ASC) VISIBLE,

```

```

162     CONSTRAINT 'management_member'
163     FOREIGN KEY ('member_id')
164     REFERENCES 'Project'.'member' ('member_id')
165     ON DELETE CASCADE
166     ON UPDATE CASCADE,
167     CONSTRAINT 'management_project'
168     FOREIGN KEY ('project_id')
169     REFERENCES 'Project'.'project' ('project_id')
170     ON DELETE CASCADE
171     ON UPDATE CASCADE)
172 ENGINE = InnoDB;
173
174
175 -----
176 -- Table 'Project'.'evaluates'
177 -----
178 CREATE TABLE IF NOT EXISTS 'Project'.'evaluates' (
179     'evaluation_id' INT NOT NULL AUTO_INCREMENT,
180     'project_id' INT NULL,
181     'researcher_id' INT NULL,
182     'grade' INT NULL,
183     'evaluation_date' DATE NULL,
184     PRIMARY KEY ('evaluation_id'),
185     INDEX 'evaluates_project_idx' ('project_id' ASC) VISIBLE,
186     INDEX 'evaluates_researcher_idx' ('researcher_id' ASC) VISIBLE,
187     CONSTRAINT 'evaluates_project'
188     FOREIGN KEY ('project_id')
189     REFERENCES 'Project'.'project' ('project_id')
190     ON DELETE CASCADE
191     ON UPDATE CASCADE,
192     CONSTRAINT 'evaluates_researcher'
193     FOREIGN KEY ('researcher_id')
194     REFERENCES 'Project'.'researcher' ('researcher_id')
195     ON DELETE CASCADE
196     ON UPDATE CASCADE)
197 ENGINE = InnoDB;
198
199
200 -----
201 -- Table 'Project'.'works_in_project'
202 -----
203 CREATE TABLE IF NOT EXISTS 'Project'.'works_in_project' (
204     'project_id' INT NULL,
205     'works_in_project_id' INT NOT NULL AUTO_INCREMENT,
206     'researcher_id' INT NULL,
207     PRIMARY KEY ('works_in_project_id'),
208     INDEX 'works_in_project_project_idx' ('project_id' ASC) INVISIBLE,
209     INDEX 'works_in_project_researcher_idx' ('researcher_id' ASC)
210     VISIBLE,
211     CONSTRAINT 'works_in_project_project'
212     FOREIGN KEY ('project_id')
213     REFERENCES 'Project'.'project' ('project_id')
214     ON DELETE CASCADE
215     ON UPDATE NO ACTION,

```

```

215     CONSTRAINT 'works_in_project_researcher'
216     FOREIGN KEY ('researcher_id')
217     REFERENCES 'Project'.'researcher' ('researcher_id')
218     ON DELETE CASCADE
219     ON UPDATE CASCADE)
220 ENGINE = InnoDB;
221
222
223 -----
224 -- Table 'Project'.'works_for_org'
225 -----
226 CREATE TABLE IF NOT EXISTS 'Project'.'works_for_org' (
227     'works_for_org_id' INT NOT NULL AUTO_INCREMENT,
228     'organization_id' INT NULL,
229     'researcher_id' INT NULL,
230     'hiring_date' INT NULL,
231     PRIMARY KEY ('works_for_org_id'),
232     INDEX 'works_for_org_organization_idx' (('organization_id' ASC)
233     VISIBLE,
234     INDEX 'works_for_org_researcher_idx' (('researcher_id' ASC) VISIBLE,
235     CONSTRAINT 'works_for_org_org'
236     FOREIGN KEY ('organization_id')
237     REFERENCES 'Project'.'organization' ('organization_id')
238     ON DELETE CASCADE
239     ON UPDATE CASCADE,
240     CONSTRAINT 'works_for_org_researcher'
241     FOREIGN KEY ('researcher_id')
242     REFERENCES 'Project'.'researcher' ('researcher_id')
243     ON DELETE CASCADE
244     ON UPDATE CASCADE)
245 ENGINE = InnoDB;
246
247 -----
248 -- Table 'Project'.'years'
249 -----
250 CREATE TABLE IF NOT EXISTS 'Project'.'years' (
251     'year_id' DATE NOT NULL,
252     PRIMARY KEY ('year_id'))
253 ENGINE = InnoDB;

```

Σχόλια:

- (α') Στους οργανισμούς αποφασίστηκε ότι αρκεί να μπορούν να αποθηκευτούν μέχρι 3 τηλέφωνα, αν υπάρχουν μόνο 2 το 3ο αρχικοποιείται σε τιμή -1 αυτόματα.
- (β') Οι κατηγορίες οργανισμών έγιναν πεδίο του πίνακα οργανισμών, αφού η μόνη διαφορά ήταν στον τίτλο των προϋπολογισμών της κάθε κατηγορίας και στο ότι τα ερευνητικά κέντρα είχαν 2 προϋπολογισμούς. Αυτή η απαίτηση καλύφθηκε με τη δημιουργία ενός πεδίου που να υποδεικνύει το είδος του οργανισμού και ανάλογα με το είδος το δεύτερο πεδίο προϋπολογισμού είτε αρχικοποιείται σε -1 είτε παίρνει την τιμή που του δίνει ο πελάτης.
- (γ') Προσθέσαμε 2 CHECKS στον πίνακα των έργων για να σιγουρευτούμε ότι οι ημε-

ρομηνίες βγάζουν νόημα και ότι το ποσό επιχορήγησης είναι εντός των αποδεκτών ορίων.

- 1.3 Μιας και απαιτούταν CRUD για όλους τους πίνακες και μιας και για να υπάρχει απάντηση σε όλα τα ερωτήματα απαιτούταν μεγάλος αριθμός έργων και ερευνητών, αποφασίστηκε να προσθεθούν indexes σε όλα τα primary και foreign keys .
- 1.4 Για την διαχείριση και ανάπτυξη της βάσης χρησιμοποιήθηκαν το MySQL Workbench και για το στήσιμο του web server χρησιμοποιήθηκε το Flask (Python) και για τη σύνδεση της βάσης και του σέρβερ χρησιμοποιήθηκε HTML, CSS, Java Script. Για την επικοινωνία frontend -backend χρησιμοποιούνται μέθοδοι GET και POST .
- 1.5 Τα βήματα εγκατάστασης είναι:
  - 1 Εγκατάσταση MySQL Workbench
  - 2 Φτιάχνουμε νέα σύνδεση στο MySQL Workbench με τα στοιχεία που επιθυμούμε και τρέχουμε το αρχείο Tables\_2.sql
  - 3 Εγκαθιστούμε τη βιβλιοθήκη Faker (Python) και το MySQL connector για Python και τρέχουμε το αρχείο generator.py αντικαθιστώντας τα στοιχεία της συνδεσής μας στο MySQL Workbench.
  - 4 Εγκαθιστούμε τη βιβλιοθήκη Flask(Python) μέσω της εντολής pip install Flask και αντικαθιστούμε τα στοιχεία της σύνδεσής μας για το MySQL Workbench στο αρχείο \_\_init\_\_.py . Τέλος τρέχουμε το main.py και συνδεόμαστε στο localhost για να δούμε την σελίδα της βάσης.