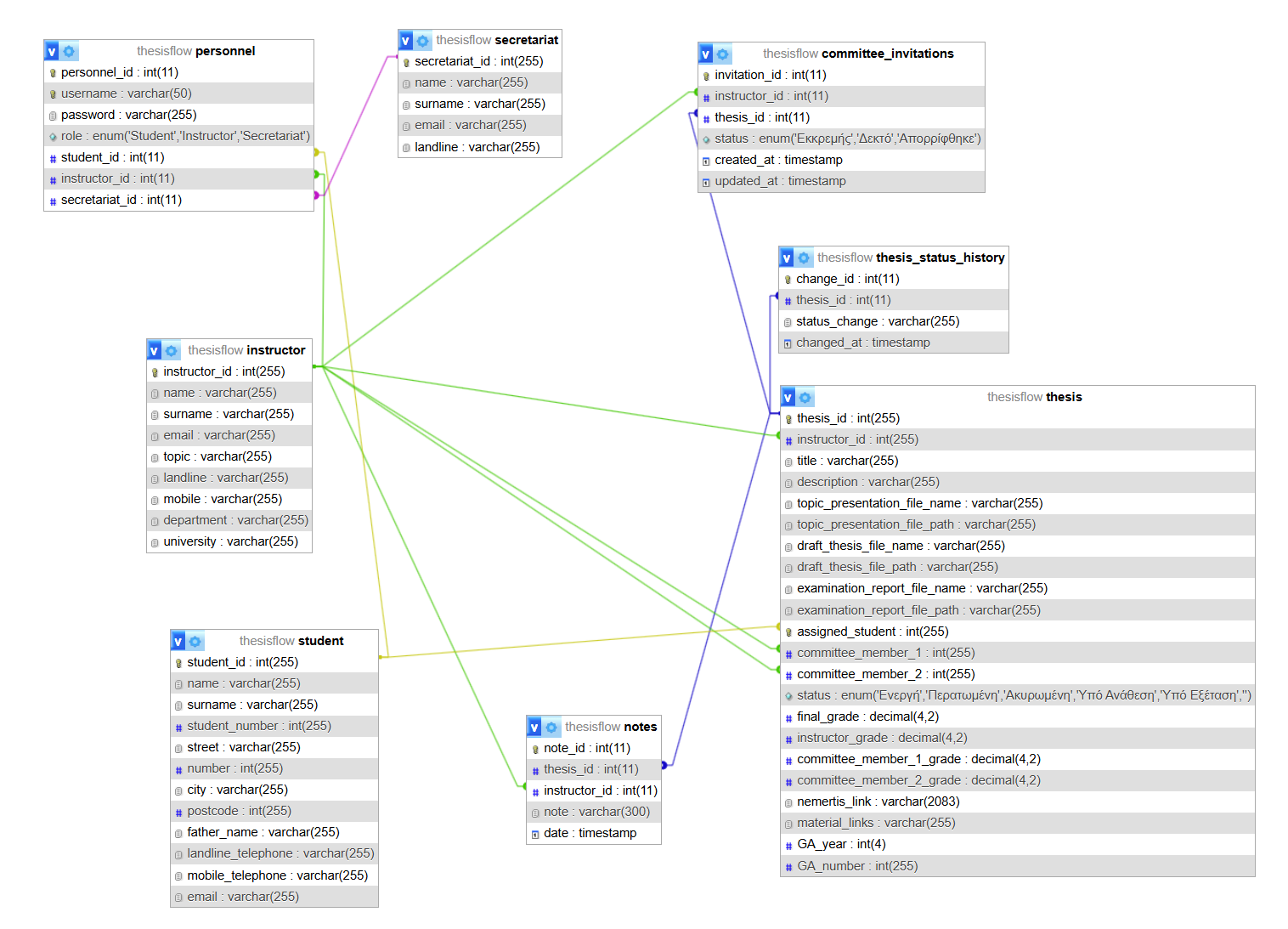


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Εργαστηριακή Άσκηση Ακαδημαϊκού Έτους 2024 - 202522.01.2025 |  | | |
|  |  |  | |  | |
|  |  | Σύστημα Υποστήριξης Διπλωματικών Εργασιών | |  | |
|  |  | | | |  |
|  | Ονοματεπώνυμο: ΑΜ:  Ονοματεπώνυμο: ΑΜ:  Ονοματεπώνυμο: ΑΜ: | | | |  |

**Σχεσιακό Διάγραμμα**

****

**Ροή Παραδείγματος**

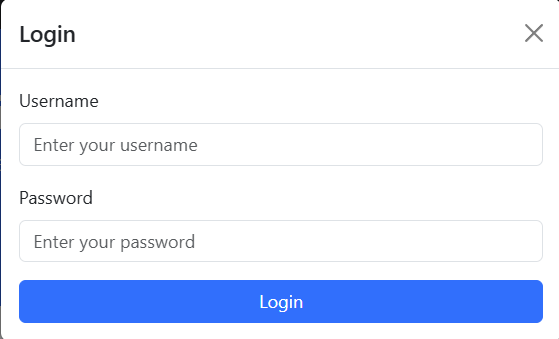
1. **Ο διδάσκων** συνδέεται, δημιουργεί ένα νέο θέμα διπλωματικής εργασίας και το αναθέτει σε έναν φοιτητή.
2. **Ο φοιτητής** συνδέεται, εξετάζει το θέμα και καλεί άλλους διδάσκοντες να σχηματίσουν την τριμελή επιτροπή.
3. Οι προσκεκλημένοι **διδάσκοντες** αποδέχονται ή απορρίπτουν τις προσκλήσεις. Μόλις δύο αποδεχτούν, η διπλωματική εργασία γίνετε «Ενεργή».
4. Ο **Φοιτητής** εργάζεται πάνω στη διπλωματική εργασία, ανεβάζοντας προσχέδια και προγραμματίζοντας την υπεράσπισή της.
5. Ο **Διδάσκων** και τα άλλα μέλη της επιτροπής εξετάζουν το προσχέδιο, καταγράφουν βαθμούς και οριστικοποιούν τη διπλωματική εργασία.
6. Η **Γραμματεία** ελέγχει και εγκρίνει την ολοκλήρωση, σημειώνοντας τη διπλωματική εργασία ως «Ολοκληρώθηκε».

**Παρουσίαση της εφαρμογής και στιγμιότυπα οθόνης**

Παρακάτω παρουσιάζονται οι κύριες ροές χρηστών με ενδεικτικά «στιγμιότυπα οθόνης» που δείχνουν πώς αλληλοεπιδρά κάθε ρόλος με το σύστημα.

**3.1. Οθόνη σύνδεση**

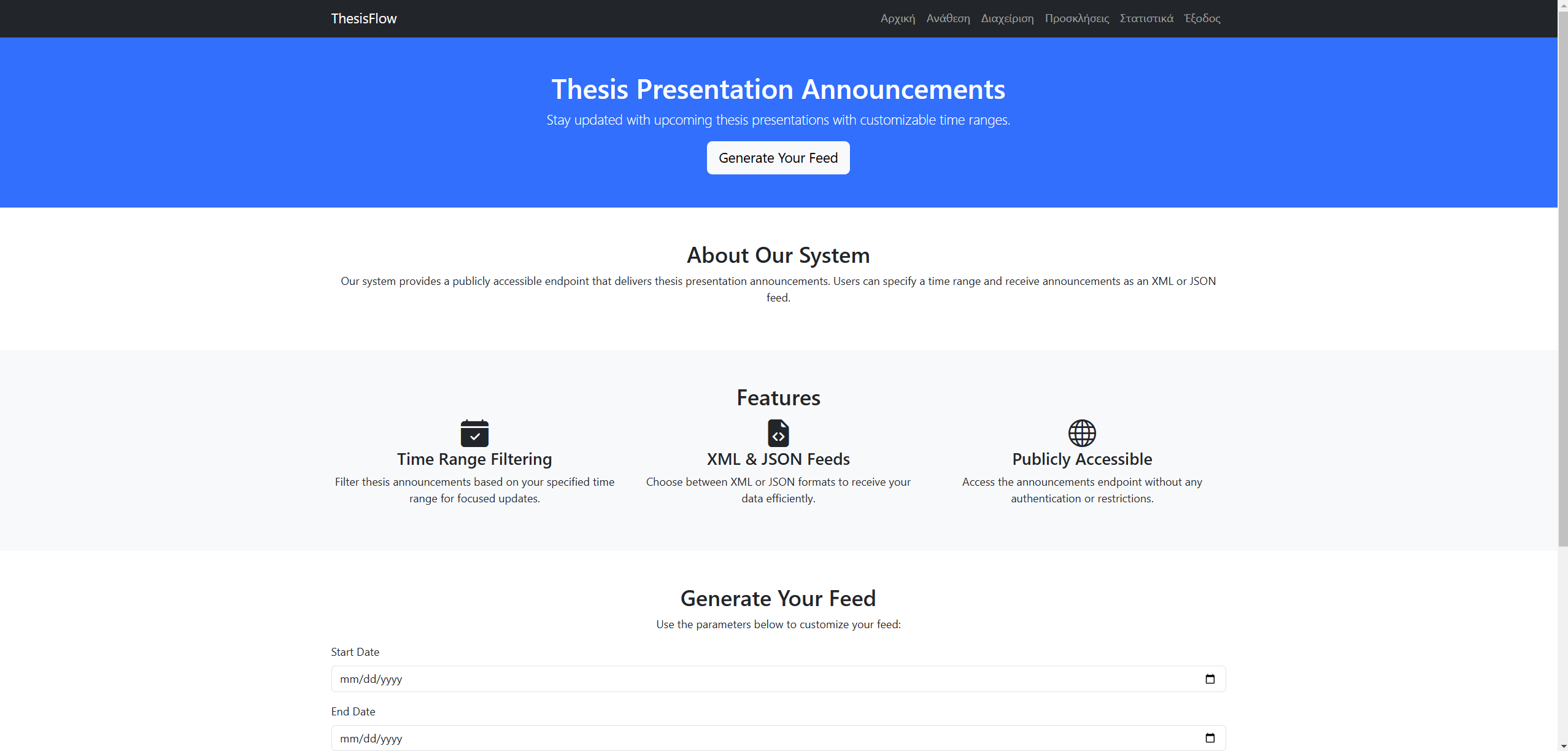
1. **Ο χρήστης επιχειρεί να αποκτήσει πρόσβαση** σε οποιαδήποτε σελίδα του συστήματος.
2. Εάν δεν πιστοποιηθεί, το σύστημα τον ανακατευθύνει στην οθόνη σύνδεσης.



* **Πεδία**: Όνομα χρήστη, κωδικός πρόσβασης
* **Κουμπιά**: «Σύνδεση»
* **Συμπεριφορά**: Μετά την επιτυχή είσοδο, ο χρήστης οδηγείται στην αντίστοιχη αρχική οθόνη. Σε περίπτωση αποτυχίας, εμφανίζεται σφάλμα.

**3.2. Αρχική οθόνη Εκπαιδευτή**

Μετά την επιτυχή είσοδο, ο ανακατευθύνεται στην **αρχική οθόνη του Εκπαιδευτή**. Εμφανίζει ένα κύριο μενού και επισημαίνει τις εκκρεμείς εργασίες του εκπαιδευτή.

****

**Επιλογές μενού**:

* 1. **Αρχική**
  2. **Δημιουργία θεμάτων διπλωματικής εργασίας και Αρχική ανάθεση σε φοιτητή**
  3. **Προβολή λίστας διπλωματικών εργασιών**
  4. **Προβολή προσκλήσεων για συμμετοχή**
  5. **Προβολή στατιστικών στοιχείων**
  6. **Έξοδος**

**3.2.1. Δημιουργία νέου θέματος διπλωματικής εργασίας**

Ο καθηγητής κάνει κλικ στην επιλογή «Δημιουργία θεμάτων διπλωματικής εργασίας και Αρχική ανάθεση σε φοιτητή» για να προσθέσει ένα νέο θέμα.

1. Πατάει το κουμπί «Προσθήκη θέματος» και ανοίγει modal για προσθήκη θέματος
2. Εισάγει τον **τίτλο**: «Σχεδιασμός ενός δικτύου αισθητήρων IoT χαμηλής ισχύος»
3. Εισάγει **σύντομη περιγραφή**: «Διερεύνηση προσεγγίσεων υλικού και λογισμικού για τη μείωση της κατανάλωσης ενέργειας σε κόμβους αισθητήρων IoT».
4. Επισυνάπτει ένα **PDF** με λεπτομερείς απαιτήσεις.
5. Κλικ **Αποθήκευση**.

A screenshot of a computer

Description automatically generated

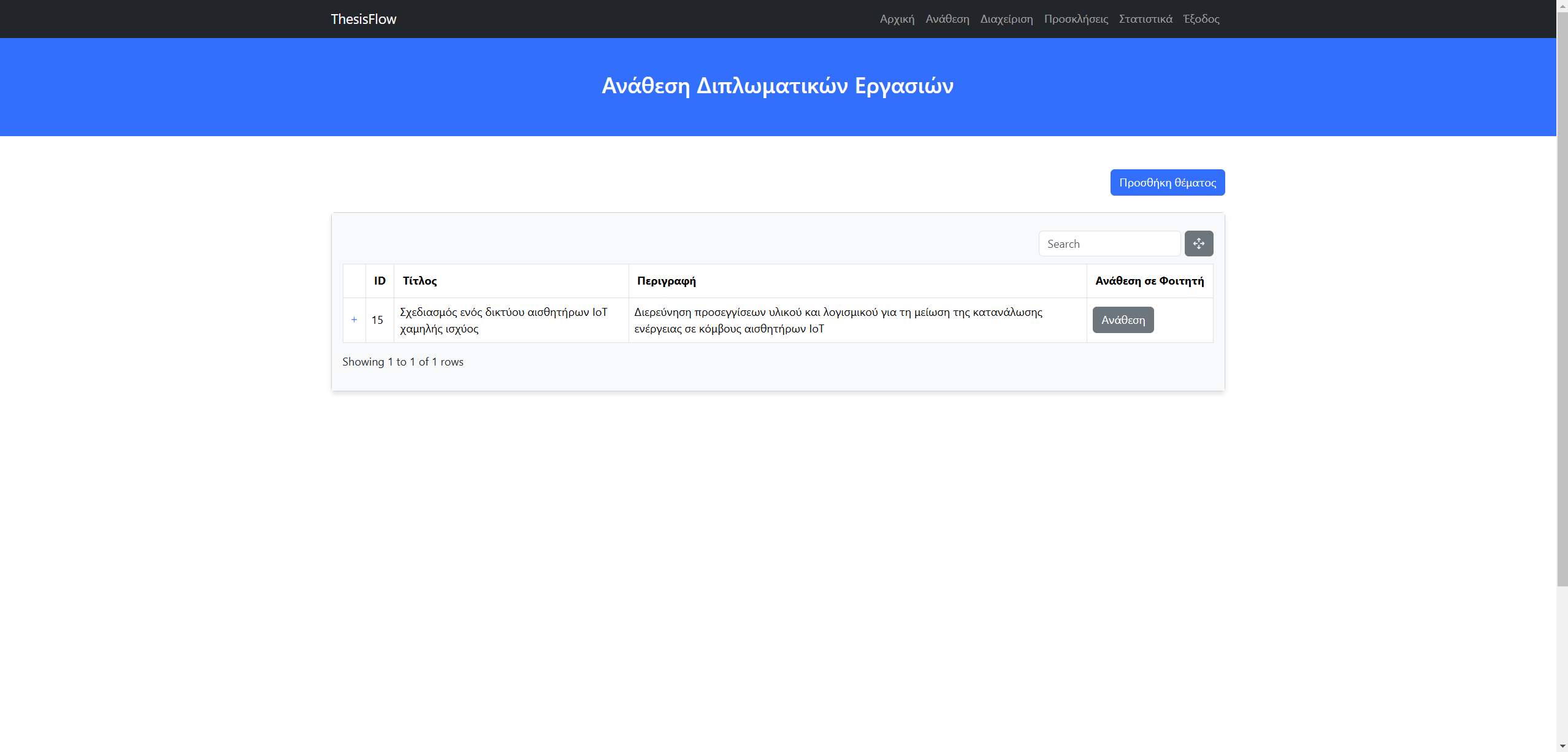
* **Πεδία**: Τίτλος, Περιγραφή, Αναλυτική Παρουσίαση Θέματος (PDF)
* **Κουμπί**: «Προσθήκη»

Ένα μήνυμα επιτυχίας επιβεβαιώνει την αποθήκευση του νέου θέματος.

**3.2.2. Αρχική ανάθεση του θέματος**

Στη συνέχεια, ο Dr. John Smith αναθέτει το νεοδημιουργηθέν θέμα στη Mary Johnson (αναγνωριστικό φοιτητή 123456).

1. Εμφανίζεται πίνακας με τις διπλωματικές χωρίς ανάθεση
2. Επιλέγει **«Ανάθεση».**
3. Πραγματοποιεί αναζήτηση με βάση το **Student ID** ή με ονοματεπωνυμο.
4. Πατάει **Ανάθεση**.

****

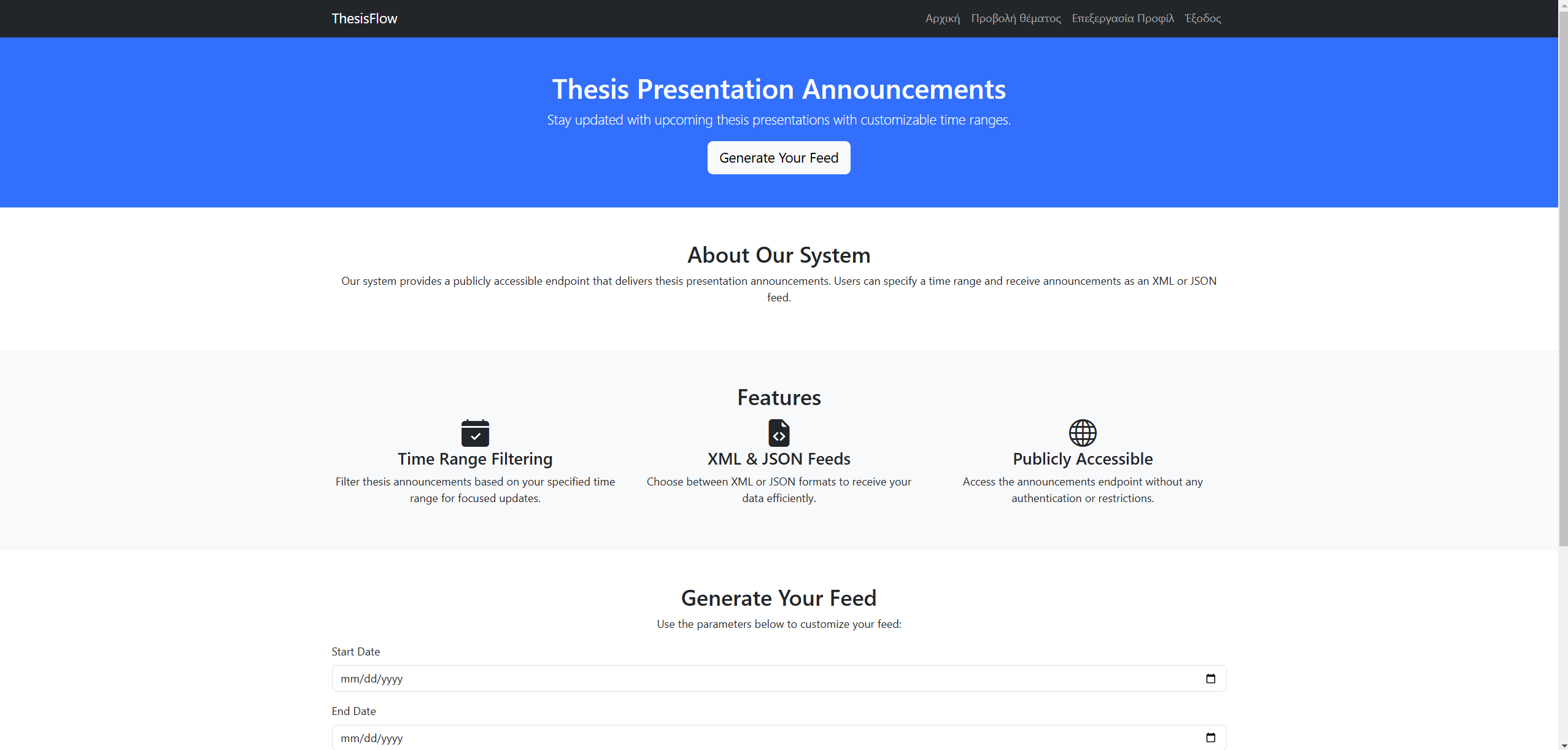
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**3.3. Αρχική οθόνη του φοιτητή**

****

* **Επιλογές μενού**:
  1. **Προβολή θέματος**
  2. **Επεξεργασία Προφίλ**

**3.3.1. Προβολή θέματος**

**A screenshot of a computer

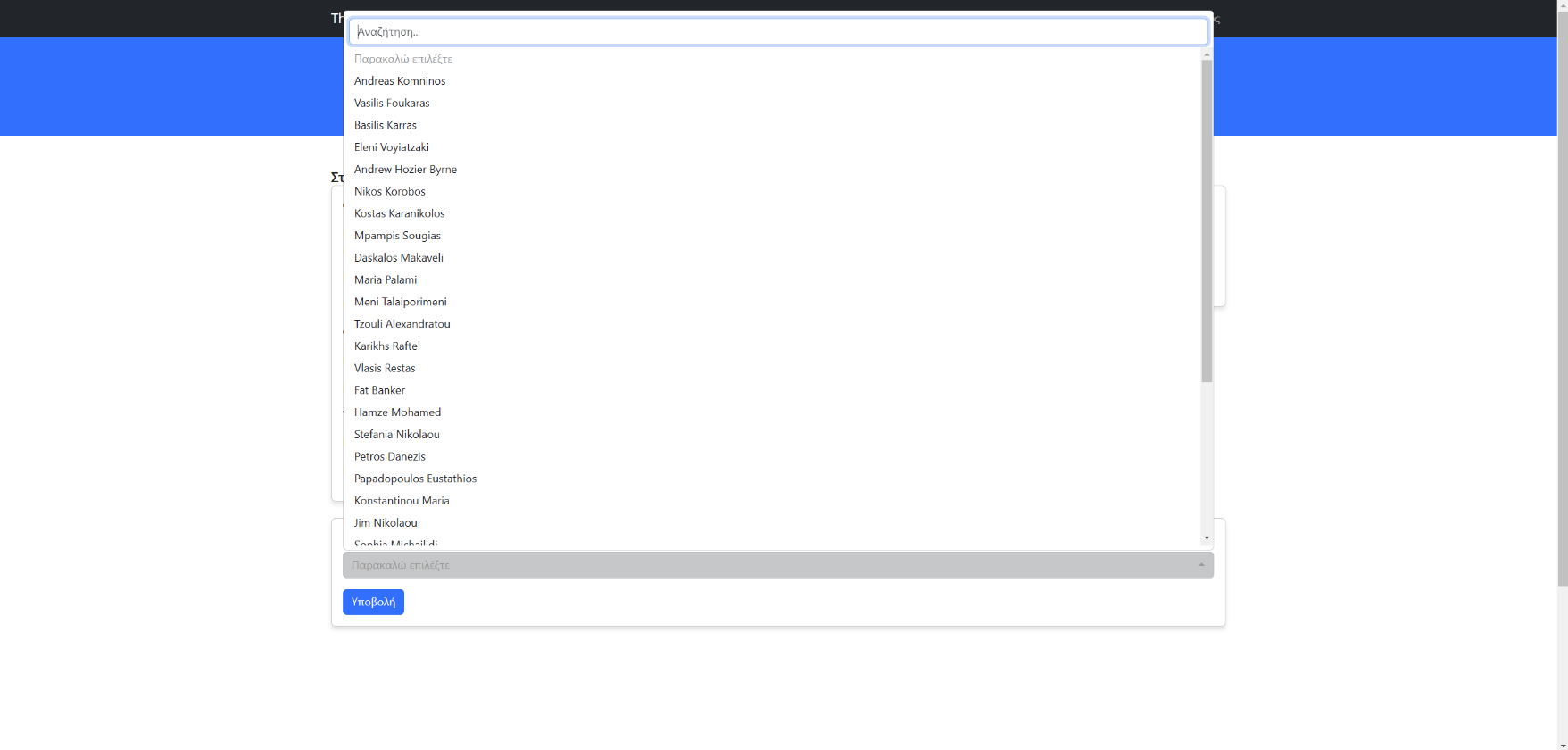
Description automatically generated**

* **Στοιχεία Πτυχιακής**
* **Μέλη Επιτροπής**
* **Επιλέξτε Καθηγητή για την Επιτροπή**

**3.3.2. Πρόσκληση μελών επιτροπής**

Για να μεταβεί από το «Υπό ανάθεση» στο «Ενεργό», η Mary πρέπει να προσκαλέσει δύο επιπλέον εκπαιδευτές.

1. Αναζητά πρόσθετους διδάσκοντες με βάση το όνομα.
2. Επιλέγει την Καθηγητές για να προσκαλέσει.

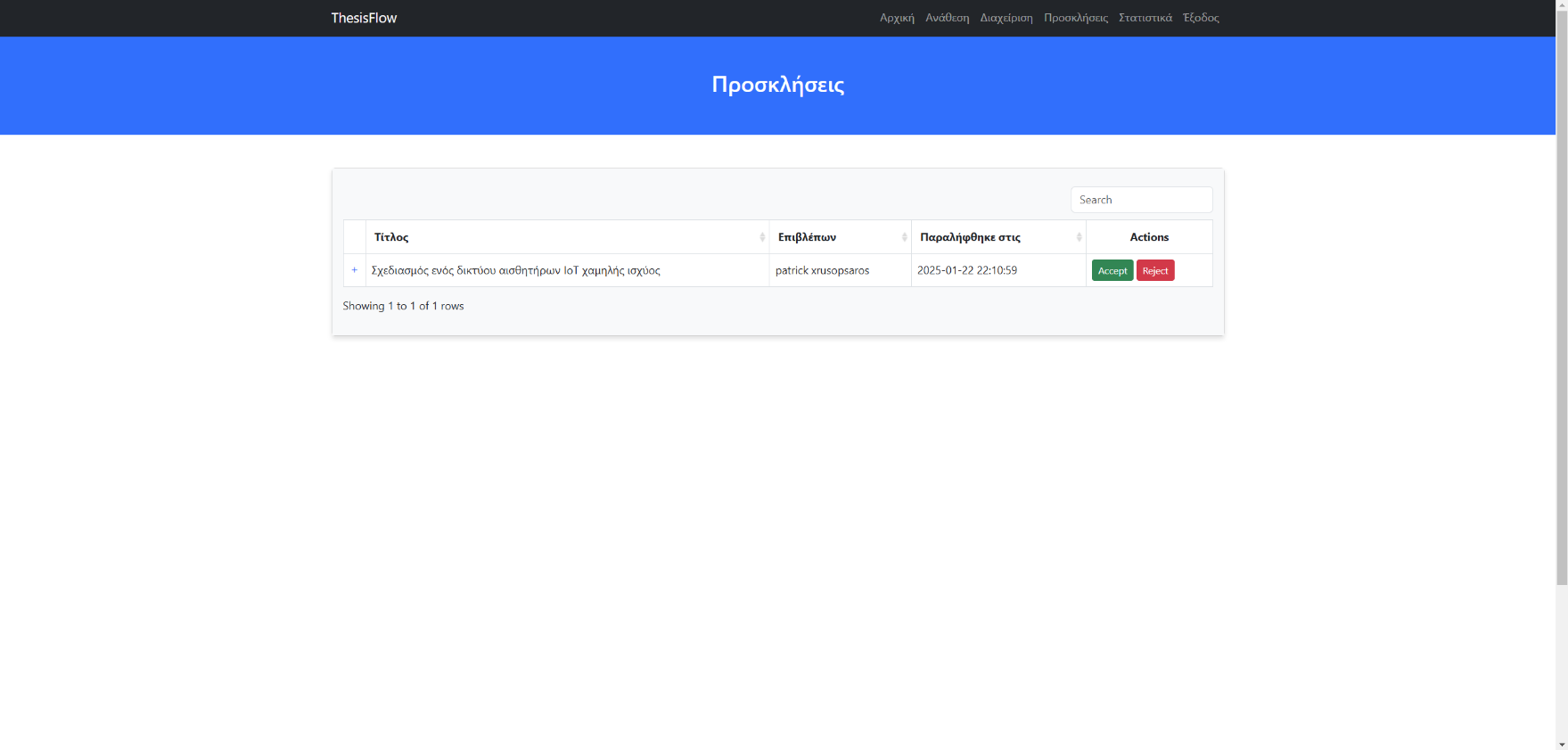
****

* **Πεδία**: Επιλέξτε Καθηγητή
* **Ενέργεια**: Υποβολή

Μόλις δύο διδάσκοντες αποδεχτούν, η διπλωματική εργασία μεταβαίνει αυτόματα σε κατάσταση **"Ενεργή** ».

**3.4. Διδάσκοντες που αποδέχονται τις προσκλήσεις για επιτροπή**

**Αρχική σελίδα του διδάσκοντος**.

****

* Εξετάζουν τις λεπτομέρειες της διατριβήq.
* Ο καθένας τους κάνει κλικ στο κουμπί **Αποδοχή** ή **Απόρριψη**.

Μετά την αποδοχή, το σύστημα ενημερώνει την τριμελή επιτροπή και η διατριβή γίνεται «ενεργή».

**3.5. Προβολή και διαχείριση ενεργών διπλωματικών εργασιών**

Τώρα που η διπλωματική εργασία είναι «Ενεργή», και τα τρία μέλη της επιτροπής μπορούν να τη δουν στη **λίστα προβολής των διπλωματικών εργασιών** τους .

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Φίλτρα**: Κατάσταση, Ρόλος

Ο καθηγητής μπορεί να κάνει κλικ στη συγκεκριμένη διπλωματική εργασία για να δει:

* **Βασικές πληροφορίες** (θέμα, φοιτητής, επιτροπή)
* **Χρονολόγιο ενεργειών** (ημερομηνίες και καταστάσεις)

**3.6. Μετάβαση σε «Υπό εξέταση» από τον επόπτη**

Όταν ο φοιτητής είναι έτοιμος να υπερασπιστεί, ο Dr. John Smith αλλάζει την κατάσταση από «Active» σε **«Under Examination».**

* **Κουμπί**: «Υπό Εξέταση»
* Προσβάσιμο μόνο από τον επιβλέποντα καθηγητή.

**3.7. Ενέργειες του φοιτητή υπό εξέταση**

Ο φοιτητής βλέπει τώρα πρόσθετα πεδία για να ανεβάσει το προσχέδιο της διπλωματικής εργασίας και να ανεβάσει το λινκ από υλικό.

1. Ανεβάζει το **PDF του σχεδίου της διπλωματικής εργασίας** και πρόσθετους **συνδέσμους** προς το αποθετήριο κώδικα/βίντεο.

A screenshot of a computer

Description automatically generated

Στη συνέχεια, το σύστημα επιτρέπει στον **επιβλέποντα** να δημιουργήσει και να εμφανίσει το επίσημο κείμενο ανακοίνωσης για την παρουσίαση της διπλωματικής εργασίας.

**3.8. Βαθμολόγηση από την επιτροπή**

Μόλις ολοκληρωθεί η παρουσίαση (εκτός συστήματος), κάθε διδάσκων καταγράφει τον βαθμό του:

1. Ο Επιβλέπων συνδέεται, ανοίγει τη διπλωματική εργασία **"Υπό εξέταση**».
2. Εμφανίζονται τα κριτήρια βαθμολόγησης.

A screenshot of a computer

Description automatically generated

Κάθε μέλος μπορεί να δει τους καταγεγραμμένους βαθμούς των άλλων.

**3.9. Διαχείριση Γραμματείας**

**A screenshot of a computer

Description automatically generated**

A screenshot of a computer

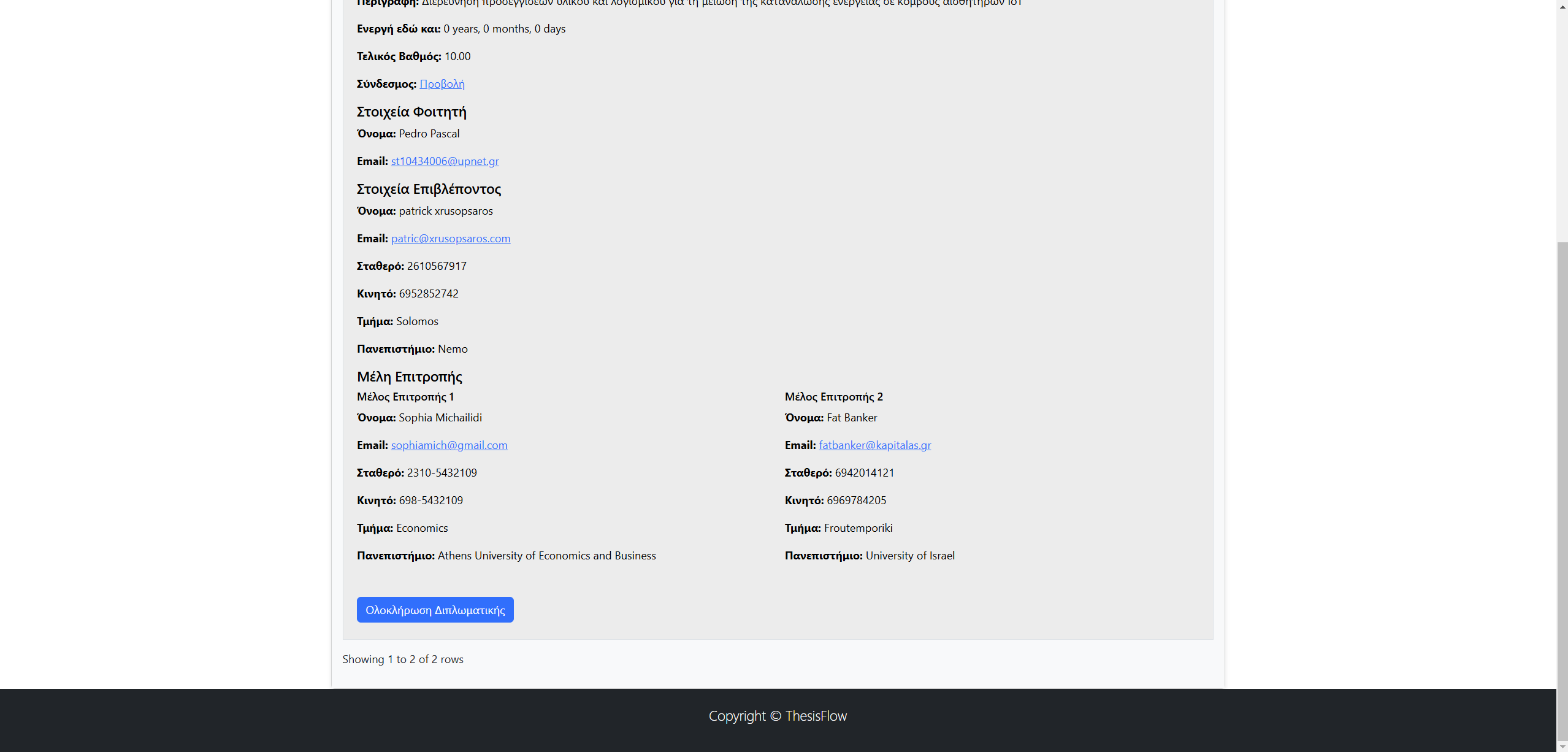
Description automatically generatedΑφού όλα τα μέλη της επιτροπής έχουν καταγράψει τους βαθμούς:

1. Ο φοιτητής βλέπει τον υπολογισμένο **τελικό βαθμό** και μπορεί πλέον να παρέχει τον **σύνδεσμο αποθετηρίου Nemertis** για το τελικό κείμενο.

A screenshot of a computer

Description automatically generated

1. Η Γραμματεία εξετάζει τη διπλωματική εργασία στην **αρχική οθόνη της Γραμματείας** στην ενότητα «Υπό εξέταση» και βλέπει ότι πληρούνται όλες οι προϋποθέσεις.
2. Η Γραμματεία κάνει κλικ στο **"Ολοκλήρωση Διπλωματικής**» για να οριστικοποιήσει τη διπλωματική εργασία.

****

**3.10. Ολοκληρωμένη διατριβή**

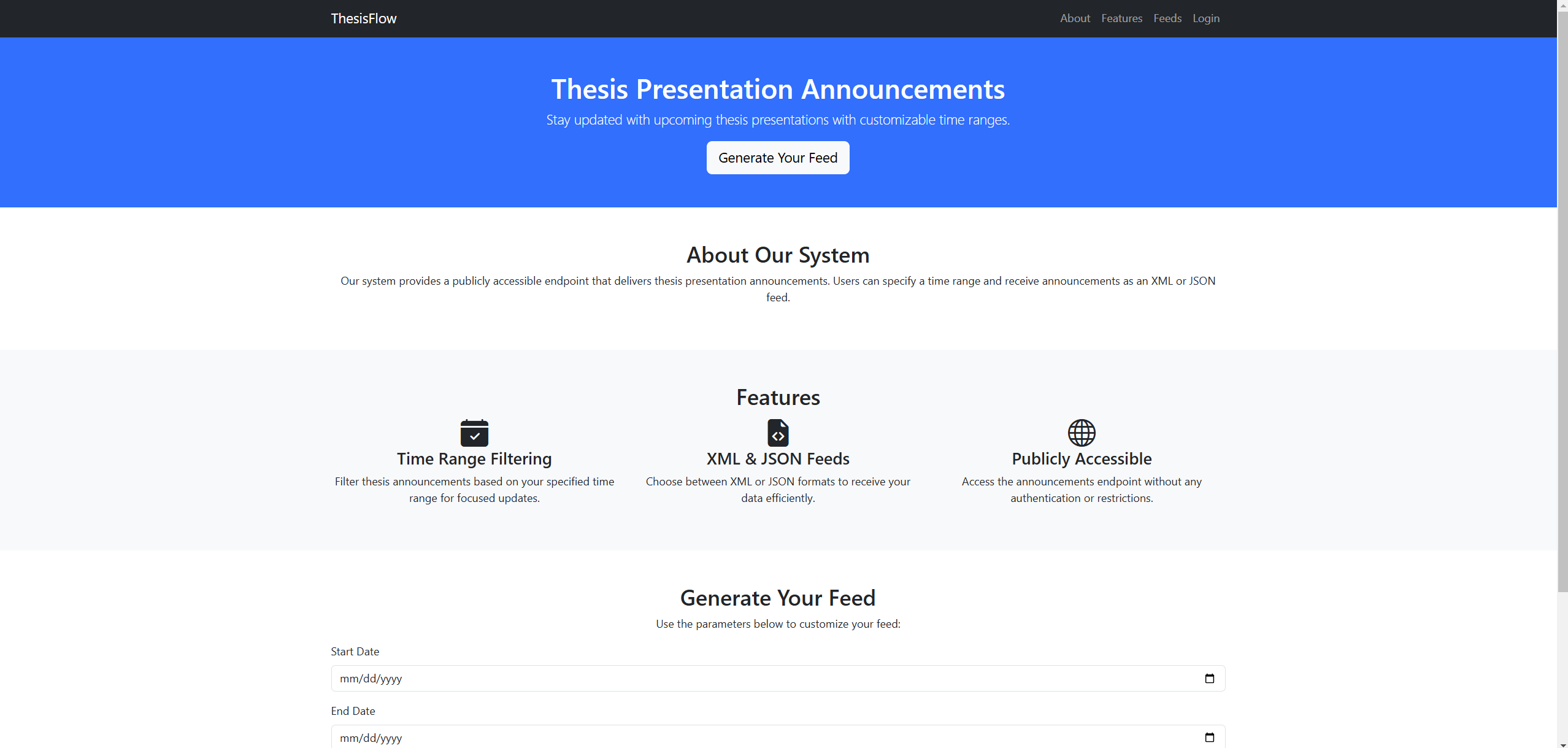
Μόλις ολοκληρωθεί:

* Ο **φοιτητής** δεν μπορεί πλέον να επεξεργαστεί τη διπλωματική εργασία. Βλέπει μόνο την τελική κατάσταση, τους καταγεγραμμένους βαθμούς και οποιαδήποτε τελική τεκμηρίωση.
* **Οι διδάσκοντες** τη βλέπουν στη λίστα «Ολοκληρώθηκε» με τον τελικό βαθμό και την αρχειοθετημένη κατάσταση.

**4. Πρόσθετες λειτουργίες του συστήματος**

**4.1. Τελικό σημείο δημόσιων ανακοινώσεων**

Το σύστημα παρέχει ένα **δημόσια προσβάσιμο τελικό σημείο** για ανακοινώσεις παρουσίασης διπλωματικών εργασιών. Οι χρήστες μπορούν να ανακτήσουν επερχόμενες ή παρελθούσες ανακοινώσεις σε μορφή **XML ή JSON** καθορίζοντας παραμέτρους ημερομηνίας. Αυτή η λειτουργία δεν απαιτεί έλεγχο ταυτότητας.



**4.2. Εισαγωγή δεδομένων (Γραμματεία)**

Οι χρήστες της Γραμματείας διαθέτουν μια λειτουργία **εισαγωγής δεδομένων** για μαζική φόρτωση διδασκόντων και φοιτητών από ένα αρχείο JSON. Αυτό είναι χρήσιμο για τον αρχικό πληθυσμό ή τις ετήσιες ενημερώσεις της βάσης δεδομένων.

A screenshot of a computer

Description automatically generated

**4.3. Στατιστικά στοιχεία εκπαιδευτών**

Οι διδάσκοντες μπορούν να βλέπουν στατιστικά στοιχεία σχετικά με τις επιβλεπόμενες και τις εμπλεκόμενες από την επιτροπή διπλωματικές εργασίες τους:

**A screenshot of a computer

Description automatically generated**

SQL Dump βάσης

-- phpMyAdmin SQL Dump

-- version 5.2.1

-- https://www.phpmyadmin.net/

--

-- Host: 127.0.0.1

-- Generation Time: Jan 22, 2025 at 09:44 PM

-- Server version: 10.4.32-MariaDB

-- PHP Version: 8.2.12

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

START TRANSACTION;

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8mb4 \*/;

--

-- Database: `thesisflow`

--

-- --------------------------------------------------------

--

-- Table structure for table `committee\_invitations`

--

CREATE TABLE `committee\_invitations` (

  `invitation\_id` int(11) NOT NULL,

  `instructor\_id` int(11) NOT NULL,

  `thesis\_id` int(11) NOT NULL,

  `status` enum('Εκκρεμής','Δεκτό','Απορρίφθηκε') NOT NULL DEFAULT 'Εκκρεμής',

  `created\_at` timestamp NOT NULL DEFAULT current\_timestamp(),

  `updated\_at` timestamp NULL DEFAULT NULL ON UPDATE current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- --------------------------------------------------------

--

-- Table structure for table `instructor`

--

CREATE TABLE `instructor` (

  `instructor\_id` int(255) NOT NULL,

  `name` varchar(255) NOT NULL,

  `surname` varchar(255) NOT NULL,

  `email` varchar(255) NOT NULL,

  `topic` varchar(255) NOT NULL,

  `landline` varchar(255) NOT NULL,

  `mobile` varchar(255) NOT NULL,

  `department` varchar(255) NOT NULL,

  `university` varchar(255) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- --------------------------------------------------------

--

-- Table structure for table `notes`

--

CREATE TABLE `notes` (

  `note\_id` int(11) NOT NULL,

  `thesis\_id` int(11) NOT NULL,

  `instructor\_id` int(11) NOT NULL,

  `note` varchar(300) NOT NULL,

  `date` timestamp NOT NULL DEFAULT current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- --------------------------------------------------------

--

-- Table structure for table `personnel`

--

CREATE TABLE `personnel` (

  `personnel\_id` int(11) NOT NULL,

  `username` varchar(50) NOT NULL,

  `password` varchar(255) NOT NULL,

  `role` enum('Student','Instructor','Secretariat') NOT NULL,

  `student\_id` int(11) DEFAULT NULL,

  `instructor\_id` int(11) DEFAULT NULL,

  `secretariat\_id` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- --------------------------------------------------------

--

-- Table structure for table `secretariat`

--

CREATE TABLE `secretariat` (

  `secretariat\_id` int(255) NOT NULL,

  `name` varchar(255) NOT NULL,

  `surname` varchar(255) NOT NULL,

  `email` varchar(255) NOT NULL,

  `landline` varchar(255) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- --------------------------------------------------------

--

-- Table structure for table `student`

--

CREATE TABLE `student` (

  `student\_id` int(255) NOT NULL,

  `name` varchar(255) NOT NULL,

  `surname` varchar(255) NOT NULL,

  `student\_number` int(255) NOT NULL,

  `street` varchar(255) NOT NULL,

  `number` int(255) NOT NULL,

  `city` varchar(255) NOT NULL,

  `postcode` int(255) NOT NULL,

  `father\_name` varchar(255) NOT NULL,

  `landline\_telephone` varchar(255) NOT NULL,

  `mobile\_telephone` varchar(255) NOT NULL,

  `email` varchar(255) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

-- --------------------------------------------------------

--

-- Table structure for table `thesis`

--

CREATE TABLE `thesis` (

  `thesis\_id` int(255) NOT NULL,

  `instructor\_id` int(255) NOT NULL,

  `title` varchar(255) NOT NULL,

  `description` varchar(255) NOT NULL,

  `topic\_presentation\_file\_name` varchar(255) DEFAULT NULL,

  `topic\_presentation\_file\_path` varchar(255) DEFAULT NULL,

  `draft\_thesis\_file\_name` varchar(255) DEFAULT NULL,

  `draft\_thesis\_file\_path` varchar(255) DEFAULT NULL,

  `examination\_report\_file\_name` varchar(255) DEFAULT NULL,

  `examination\_report\_file\_path` varchar(255) DEFAULT NULL,

  `assigned\_student` int(255) DEFAULT NULL,

  `committee\_member\_1` int(255) DEFAULT NULL,

  `committee\_member\_2` int(255) DEFAULT NULL,

  `status` enum('Ενεργή','Περατωμένη','Ακυρωμένη','Υπό Ανάθεση','Υπό Εξέταση','') NOT NULL DEFAULT 'Υπό Ανάθεση',

  `final\_grade` decimal(4,2) DEFAULT NULL CHECK (`final\_grade` between 1 and 10),

  `instructor\_grade` decimal(4,2) DEFAULT NULL CHECK (`instructor\_grade` between 1 and 10),

  `committee\_member\_1\_grade` decimal(4,2) DEFAULT NULL CHECK (`committee\_member\_1\_grade` between 1 and 10),

  `committee\_member\_2\_grade` decimal(4,2) DEFAULT NULL CHECK (`committee\_member\_2\_grade` between 1 and 10),

  `nemertis\_link` varchar(2083) DEFAULT NULL,

  `material\_links` varchar(255) DEFAULT NULL,

  `GA\_year` int(4) DEFAULT NULL,

  `GA\_number` int(255) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

--

-- Triggers `thesis`

--

DELIMITER $$

CREATE TRIGGER `thesis\_insert\_trigger` AFTER INSERT ON `thesis` FOR EACH ROW BEGIN

    -- Insert a record into thesis\_status\_history for new thesis creation

    INSERT INTO thesis\_status\_history (thesis\_id, status\_change, changed\_at)

    VALUES (NEW.thesis\_id, 'Δημιουργήθηκε', NOW());

END

$$

DELIMITER ;

DELIMITER $$

CREATE TRIGGER `thesis\_status\_change\_trigger` AFTER UPDATE ON `thesis` FOR EACH ROW BEGIN

    DECLARE status\_change VARCHAR(255);

    -- Determine the details of the change

    IF OLD.status <> NEW.status THEN

      SET status\_change = NEW.status;

    ELSEIF (NEW.description IS NOT NULL AND OLD.description IS NULL) OR OLD.description <> NEW.description THEN

      SET status\_change = 'Περιγραφή Ενημερώθηκε';

    ELSEIF (NEW.topic\_presentation\_file\_name IS NOT NULL AND OLD.topic\_presentation\_file\_name IS NULL) OR OLD.topic\_presentation\_file\_name <> NEW.topic\_presentation\_file\_name THEN

      SET status\_change = 'Αναλυτική Παρουσίαση Ενημερώθηκε';

    ELSEIF (NEW.draft\_thesis\_file\_name IS NOT NULL AND OLD.draft\_thesis\_file\_name IS NULL) OR OLD.draft\_thesis\_file\_name <> NEW.draft\_thesis\_file\_name THEN

      SET status\_change = 'Πρόχειρο Κείμενο Διπλωματικής Ενημερώθηκε';

    ELSEIF (NEW.examination\_report\_file\_name IS NOT NULL AND OLD.examination\_report\_file\_name IS NULL) OR OLD.examination\_report\_file\_name <> NEW.examination\_report\_file\_name THEN

      SET status\_change = 'Πρακτικού Εξέτασης Ενημερώθηκε';

    ELSEIF (NEW.assigned\_student IS NOT NULL AND OLD.assigned\_student IS NULL) THEN

      SET status\_change = 'Ανατέθηκε';

    ELSEIF (NEW.assigned\_student IS NULL AND OLD.assigned\_student IS NOT NULL) THEN

      SET status\_change = 'Ανάθεση Αφαιρέθηκε';

    ELSEIF OLD.assigned\_student <> NEW.assigned\_student THEN

      SET status\_change = 'Ανάθεση Ενημερώθηκε';

    ELSEIF (NEW.committee\_member\_1 IS NOT NULL AND OLD.committee\_member\_1 IS NULL) THEN

      SET status\_change = 'Επιτροπή Ενημερώθηκε';

    ELSEIF (NEW.committee\_member\_2 IS NOT NULL AND OLD.committee\_member\_2 IS NULL) THEN

      SET status\_change = 'Επιτροπή Ενημερώθηκε';

    ELSEIF OLD.committee\_member\_1 <> NEW.committee\_member\_1 OR OLD.committee\_member\_2 <> NEW.committee\_member\_2 THEN

      SET status\_change = 'Επιτροπή Ενημερώθηκε';

    ELSEIF (NEW.final\_grade IS NOT NULL AND OLD.final\_grade IS NULL) OR OLD.final\_grade <> NEW.final\_grade THEN

      SET status\_change = 'Τελικός Βαθμός Αναρτήθηκε';

    ELSEIF (NEW.instructor\_grade IS NOT NULL AND OLD.instructor\_grade IS NULL) OR OLD.instructor\_grade <> NEW.instructor\_grade THEN

      SET status\_change = 'Βαθμός Επιτηρητή Αναρτήθηκε';

    ELSEIF (NEW.committee\_member\_1\_grade IS NOT NULL AND OLD.committee\_member\_1\_grade IS NULL) OR OLD.committee\_member\_1\_grade <> NEW.committee\_member\_1\_grade THEN

      SET status\_change = 'Βαθμός Επιτροπής Αναρτήθηκε';

    ELSEIF (NEW.committee\_member\_2\_grade IS NOT NULL AND OLD.committee\_member\_2\_grade IS NULL) OR OLD.committee\_member\_2\_grade <> NEW.committee\_member\_2\_grade THEN

      SET status\_change = 'Βαθμός Επιτροπής Αναρτήθηκε';

    ELSEIF (NEW.nemertis\_link IS NOT NULL AND OLD.nemertis\_link IS NULL) OR OLD.nemertis\_link <> NEW.nemertis\_link THEN

      SET status\_change = 'Σύνδεσμος Νημερτή Ενημερώθηκε';

    ELSE

      SET status\_change = 'Άλλο';

    END IF;

    -- Insert a new record into the thesis\_status\_history table

    INSERT INTO thesis\_status\_history (thesis\_id, status\_change, changed\_at)

    VALUES (NEW.thesis\_id, status\_change, NOW());

END

$$

DELIMITER ;

-- --------------------------------------------------------

--

-- Table structure for table `thesis\_status\_history`

--

CREATE TABLE `thesis\_status\_history` (

  `change\_id` int(11) NOT NULL,

  `thesis\_id` int(11) NOT NULL,

  `status\_change` varchar(255) NOT NULL,

  `changed\_at` timestamp NOT NULL DEFAULT current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

--

-- Indexes for dumped tables

--

--

-- Indexes for table `committee\_invitations`

--

ALTER TABLE `committee\_invitations`

  ADD PRIMARY KEY (`invitation\_id`),

  ADD KEY `instructor\_id` (`instructor\_id`),

  ADD KEY `thesis\_id` (`thesis\_id`);

--

-- Indexes for table `instructor`

--

ALTER TABLE `instructor`

  ADD PRIMARY KEY (`instructor\_id`);

--

-- Indexes for table `notes`

--

ALTER TABLE `notes`

  ADD PRIMARY KEY (`note\_id`),

  ADD KEY `fk\_notes\_thesis` (`thesis\_id`),

  ADD KEY `fk\_notes\_instructor` (`instructor\_id`);

--

-- Indexes for table `personnel`

--

ALTER TABLE `personnel`

  ADD PRIMARY KEY (`personnel\_id`),

  ADD UNIQUE KEY `username` (`username`),

  ADD KEY `student\_id` (`student\_id`),

  ADD KEY `instructor\_id` (`instructor\_id`),

  ADD KEY `secretariat\_id` (`secretariat\_id`);

--

-- Indexes for table `secretariat`

--

ALTER TABLE `secretariat`

  ADD PRIMARY KEY (`secretariat\_id`);

--

-- Indexes for table `student`

--

ALTER TABLE `student`

  ADD PRIMARY KEY (`student\_id`);

--

-- Indexes for table `thesis`

--

ALTER TABLE `thesis`

  ADD PRIMARY KEY (`thesis\_id`),

  ADD UNIQUE KEY `assigned\_student` (`assigned\_student`),

  ADD KEY `instructor\_id` (`instructor\_id`),

  ADD KEY `fk\_committee\_member\_1` (`committee\_member\_1`),

  ADD KEY `fk\_committee\_member\_2` (`committee\_member\_2`);

--

-- Indexes for table `thesis\_status\_history`

--

ALTER TABLE `thesis\_status\_history`

  ADD PRIMARY KEY (`change\_id`),

  ADD KEY `thesis\_id` (`thesis\_id`);

--

-- AUTO\_INCREMENT for dumped tables

--

--

-- AUTO\_INCREMENT for table `committee\_invitations`

--

ALTER TABLE `committee\_invitations`

  MODIFY `invitation\_id` int(11) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `instructor`

--

ALTER TABLE `instructor`

  MODIFY `instructor\_id` int(255) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `notes`

--

ALTER TABLE `notes`

  MODIFY `note\_id` int(11) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `personnel`

--

ALTER TABLE `personnel`

  MODIFY `personnel\_id` int(11) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `secretariat`

--

ALTER TABLE `secretariat`

  MODIFY `secretariat\_id` int(255) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `student`

--

ALTER TABLE `student`

  MODIFY `student\_id` int(255) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `thesis`

--

ALTER TABLE `thesis`

  MODIFY `thesis\_id` int(255) NOT NULL AUTO\_INCREMENT;

--

-- AUTO\_INCREMENT for table `thesis\_status\_history`

--

ALTER TABLE `thesis\_status\_history`

  MODIFY `change\_id` int(11) NOT NULL AUTO\_INCREMENT;

--

-- Constraints for dumped tables

--

--

-- Constraints for table `committee\_invitations`

--

ALTER TABLE `committee\_invitations`

  ADD CONSTRAINT `committee\_invitations\_ibfk\_1` FOREIGN KEY (`instructor\_id`) REFERENCES `instructor` (`instructor\_id`) ON DELETE CASCADE,

  ADD CONSTRAINT `committee\_invitations\_ibfk\_2` FOREIGN KEY (`thesis\_id`) REFERENCES `thesis` (`thesis\_id`) ON DELETE CASCADE;

--

-- Constraints for table `notes`

--

ALTER TABLE `notes`

  ADD CONSTRAINT `fk\_notes\_instructor` FOREIGN KEY (`instructor\_id`) REFERENCES `instructor` (`instructor\_id`) ON DELETE CASCADE,

  ADD CONSTRAINT `fk\_notes\_thesis` FOREIGN KEY (`thesis\_id`) REFERENCES `thesis` (`thesis\_id`) ON DELETE CASCADE;

--

-- Constraints for table `personnel`

--

ALTER TABLE `personnel`

  ADD CONSTRAINT `personnel\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `student` (`student\_id`) ON DELETE SET NULL,

  ADD CONSTRAINT `personnel\_ibfk\_2` FOREIGN KEY (`instructor\_id`) REFERENCES `instructor` (`instructor\_id`) ON DELETE SET NULL,

  ADD CONSTRAINT `personnel\_ibfk\_3` FOREIGN KEY (`secretariat\_id`) REFERENCES `secretariat` (`secretariat\_id`) ON DELETE SET NULL;

--

-- Constraints for table `thesis`

--

ALTER TABLE `thesis`

  ADD CONSTRAINT `fk\_committee\_member\_1` FOREIGN KEY (`committee\_member\_1`) REFERENCES `instructor` (`instructor\_id`) ON DELETE SET NULL ON UPDATE CASCADE,

  ADD CONSTRAINT `fk\_committee\_member\_2` FOREIGN KEY (`committee\_member\_2`) REFERENCES `instructor` (`instructor\_id`) ON DELETE SET NULL ON UPDATE CASCADE,

  ADD CONSTRAINT `fk\_thesis\_student` FOREIGN KEY (`assigned\_student`) REFERENCES `student` (`student\_id`) ON DELETE SET NULL ON UPDATE CASCADE,

  ADD CONSTRAINT `thesis\_ibfk\_1` FOREIGN KEY (`instructor\_id`) REFERENCES `instructor` (`instructor\_id`) ON DELETE CASCADE;

--

-- Constraints for table `thesis\_status\_history`

--

ALTER TABLE `thesis\_status\_history`

  ADD CONSTRAINT `thesis\_status\_history\_ibfk\_1` FOREIGN KEY (`thesis\_id`) REFERENCES `thesis` (`thesis\_id`) ON DELETE CASCADE;

COMMIT;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;