API LESSON SCHEDULING

GET	ServletController?action=list-users

LOGIN: TRUE
ADMIN: TRUE

HTTP 200 OK

```json
[
  {
    "id": 1,
    "name": "Raul",
    "surname": "Palade",
    "email": "raul.palade@edu.unito.it",
    "administrator": false
  }
]
```

HTTP 401 UNAUTHORIZED

GET	ServletController?action=list-teachers

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

```json
[
  {
    "id": 1,
    "name": "Luca",
    "surname": "Roversi",
    "email": "luca.roversi@unito.it"
  }
]
```

GET	ServletController?action=list-teachers-availability

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

```
[
  {
    "id": 1,
    "day": "Lunedì",
    "hour": 15
  }
]
```

GET	ServletController?action=list-courses

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

```
[
  {
    "id": 1,
    "title": "Programmazione I"
  }
]
```

GET	ServletController?action=list-teacher-courses

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

```
[
  {
    "teacher": {
      "id": 1,
      "name": "Luca",
      "surname": "Roversi",
      "email": "luca.roversi@unito.it"
    },
    "course": {
      "id": 7,
      "title": "Algoritmi e Strutture Dati"
    }
  }
]
```

GET	ServletController?action=list-bookings

LOGIN: TRUE
ADMIN: TRUE

HTTP 200 OK

```
[
  {
    "user": {
      "id": 0,
      "name": "Raul",
      "surname": "Palade",
      "email": "raul.palade@edu.unito.it",
      "administrator": false
    },
    "timeSlot": {
      "id": 0,
      "day": "Lunedì",
      "hour": 15
    },
    "teacherCourse": {
      "teacher": {
        "id": 0,
        "name": "Luca",
        "surname": "Padovani",
        "email": "luca.padovani@unito.it"
      },
      "course": {
        "id": 0,
        "title": "Algoritmi e Strutture Dati"
      }
    },
    "deleted": false,
    "completed": false
  }
]
```

HTTP 401 UNAUTHORIZED

GET	ServletController?action=list-personal-bookings

LOGIN: TRUE
ADMIN: FALSE

HTTP 200 OK

```json
[
  {
    "timeSlot": {
      "id": 0,
      "day": "Lunedì",
      "hour": 15
    },
    "teacherCourse": {
      "teacher": {
        "id": 0,
        "name": "Luca",
        "surname": "Padovani",
        "email": "luca.padovani@unito.it"
      },
      "course": {
        "id": 0,
        "title": "Algoritmi e Strutture Dati"
      }
    },
    "deleted": false,
    "completed": false
  },
```

HTTP 401 UNAUTHORIZED

GET	ServletController?action=list-teacher-by-course

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

```json
[
  {
    "id": 0,
    "name": "Luca",
    "surname": "Roversi"
  }
]
```

GET	ServletController?action=list-time-slots

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

```
[
  {
    "id": 1,
    "day": "Lunedì",
    "hour": 15
  }
]
```

HTTP 401 UNAUTHORIZED

POST	ServletController?action=login

LOGIN: FALSE
ADMIN: FALSE

Parameters: email: String, password: String

HTTP 200 OK

```
[
  {"isAdmin": true}
]
```

HTTP 401 UNAUTHORIZED

POST	ServletController?action=logout

LOGIN: FALSE
ADMIN: FALSE

HTTP 200 OK

POST	ServletController?action=insert-user

Parameters: name: String, username: String, email: String

HTTP 201 OK: when the new user is inserted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

HTTP 409 CONFLICT: when the user already exists

POST	ServletController?action=activate-user

LOGIN: TRUE
ADMIN: TRUE
Parameters: name: String, username: String, email: String

HTTP 201 OK: when the user is activated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=deactivate-user

LOGIN: TRUE
ADMIN: TRUE
Parameters: name: String, username: String, email: String

HTTP 201 OK: when the user is deactivated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=insert-teacher

LOGIN: TRUE
ADMIN: TRUE
Parameters: name: String, username: String, email: String

HTTP 201 OK: when the new user is inserted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

HTTP 409 CONFLICT: when the teacher already exists

POST	ServletController?action=activate-teacher

LOGIN: TRUE
ADMIN: TRUE
Parameters: name: String, username: String, email: String

HTTP 201 OK: when the teacher is activated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=deactivate-teacher

LOGIN: TRUE
ADMIN: TRUE
Parameters: name: String, username: String, email: String

HTTP 201 OK: when the teacher is deactivated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=insert-course

LOGIN: TRUE
ADMIN: TRUE
Parameters: title: String

HTTP 201 OK: when the new course is inserted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

HTTP 409 CONFLICT: when the course already exists

POST	ServletController?action=activate-course

LOGIN: TRUE
ADMIN: TRUE
Parameters: title: String

HTTP 201 OK: when the course is activated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=deactivate-course

LOGIN: TRUE
ADMIN: TRUE
Parameters: title: String

HTTP 201 OK: when the course is deactivated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=insert-time-slot

LOGIN: TRUE
ADMIN: TRUE
Parameters: day: String, int hour

HTTP 201 OK: when the new time-slot is inserted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

HTTP 409 CONFLICT: when the time-slot already exists

POST	ServletController?action=activate-time-slot

LOGIN: TRUE
ADMIN: TRUE
Parameters: day: String, int hour

HTTP 201 OK: when the time-slot is activated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=deactivate-time-slot

LOGIN: TRUE
ADMIN: TRUE
Parameters: day: String, int hour

HTTP 201 OK: when the time-slot is deactivated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=assign-teaching

LOGIN: TRUE
ADMIN: TRUE
Parameters: teacher-email: String, title: String

HTTP 201 OK: when the new teacher-course association is inserted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

HTTP 409 CONFLICT: when the teacher-course association already exists

POST	ServletController?action=activate-teaching

LOGIN: TRUE
ADMIN: TRUE
Parameters: day: String, int hour

HTTP 201 OK: when the teacher-course association is activated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=deactivate-teaching

LOGIN: TRUE
ADMIN: TRUE
Parameters: day: String, int hour

HTTP 201 OK: when the teacher-course association is deactivated correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged and not admin

POST	ServletController?action=insert-booking

LOGIN: TRUE
ADMIN: FALSE
Parameters: user-email: String, day: String, hour: int, teacher-email: String, title: String

HTTP 201 OK: when the new booking is inserted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged

HTTP 409 CONFLICT: when the booking already exists

POST	ServletController?action=delete-booking

LOGIN: TRUE
ADMIN: FALSE
Parameters: user-email: String, day: String, hour: int, teacher-email: String, title: String

HTTP 201 OK: when the booking is deleted correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged

POST	ServletController?action=complete-booking

LOGIN: TRUE
ADMIN: FALSE
Parameters: user-email: String, day: String, hour: int, teacher-email: String, title: String

HTTP 201 OK: when the booking is marked completed correctly

HTTP 400 BAD REQUEST: when the form data is incorrect

HTTP 401 UNAUTHORIZED: when the user client is not logged