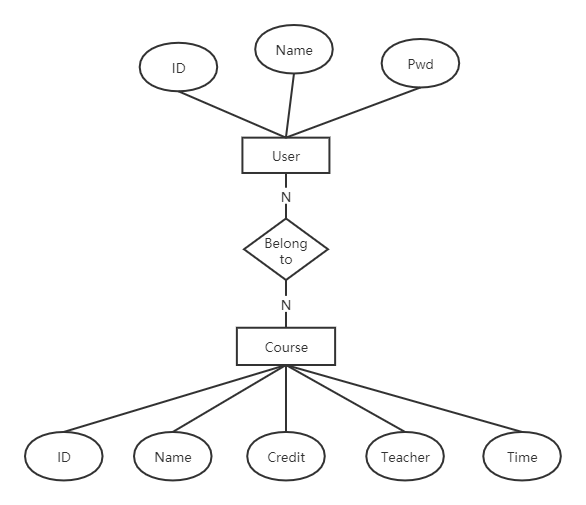
**Vue CLI** (node.js) **Spring Boot**

**IDEA, WebStorm**

1. **E-R Diagram**



1. **Log in**

Input the user information to login the system.

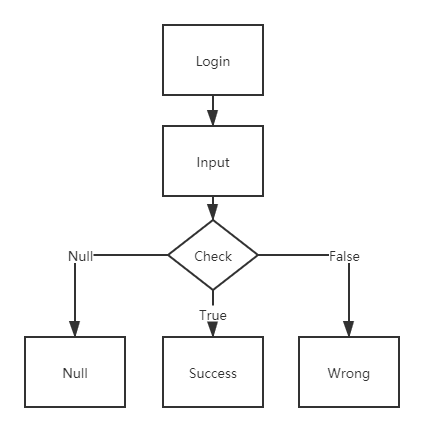
**Post:**

|  |  |
| --- | --- |
| Parameters | Meanings |
| ID | School ID |
| PWD | Password |

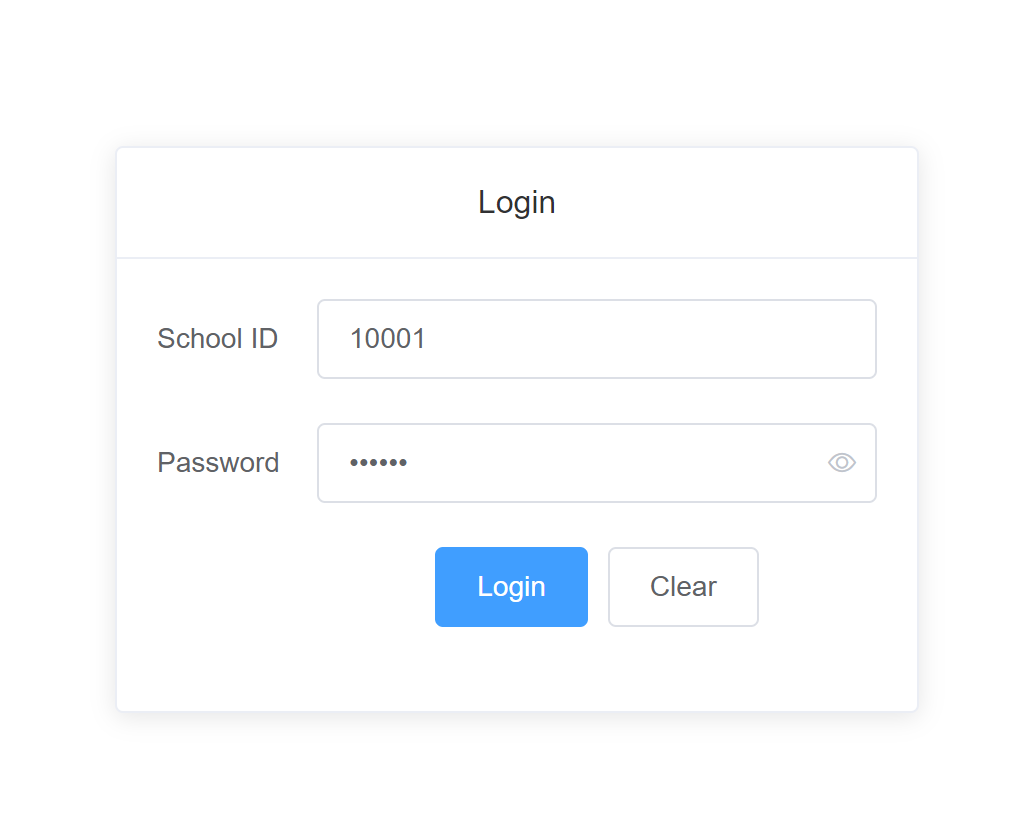
**Response:**

|  |  |
| --- | --- |
| Parameters | Meanings |
| Status | Success(exist), Wrong(error input), Null(not exit) |
| User | If Account exists, return user info |

**Flow Chart：**



**Client:**



**Server/core Code:**

@RequestMapping(value = "/login", method = RequestMethod.POST)  
public JSONObject login(@RequestBody JSONObject userInfo) {  
 int id = userInfo.getInteger("id");  
 String pwd = userInfo.getString("pwd");  
 User user = userRepository.findById(id).orElse(null);  
 JSONObject ans = new JSONObject();  
 String name = "";  
 String status = "";  
 if (user != null) {  
 if (user.getPwd().equals(pwd)) {  
 status = "success";  
 name = user.getName();  
 } else {  
 status = "wrong";  
 }  
 } else {  
 status = "null";  
 }  
 ans.put("status", status);  
 ans.put("name", name);  
 return ans;  
}

1. **Add Course**

Add Course Information to the Course Table.

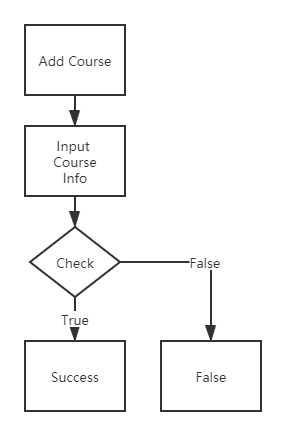
**Post:**

|  |  |
| --- | --- |
| Parameters | Meanings |
| ID | Course ID |
| Name | Course Name |
| Credit | Course Credit |
| Teacher | Course Teacher |
| Time | Start Time-End Time |

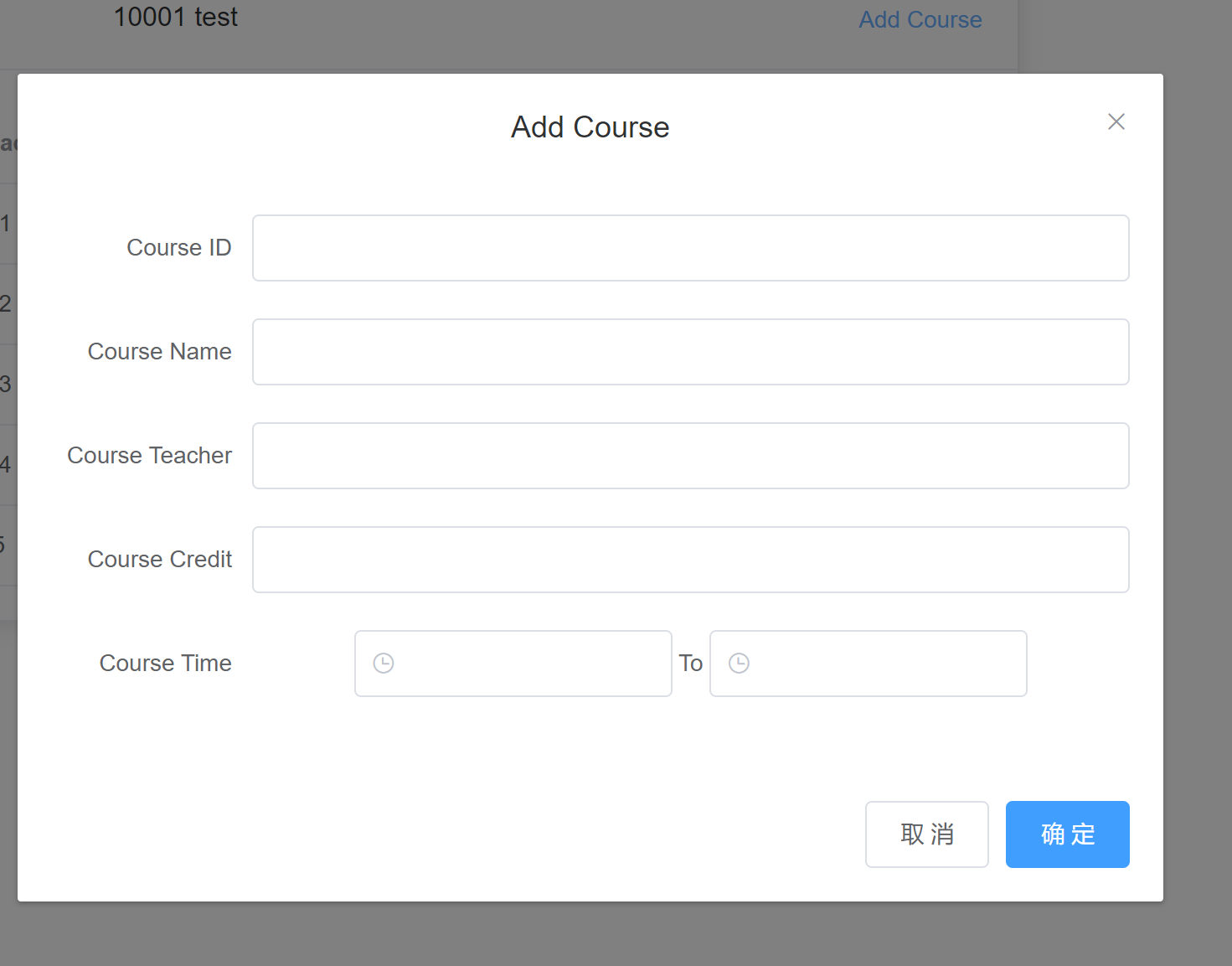
**Response:**

|  |  |
| --- | --- |
| Parameters | Meanings |
| Status | True or False |

**Flow Chart:**



**Client:**



**Server Code:**

@RequestMapping(value = "/addCourse", method = RequestMethod.POST)  
public boolean addCourse(@RequestBody JSONObject classInfo) {  
 int id = classInfo.getInteger("id");  
 String name = classInfo.getString("name");  
 String teacher = classInfo.getString("teacher");  
 int credit = classInfo.getInteger("credit");  
 String time = classInfo.getString("time");  
 Course course = new Course();  
 course.setId(id);  
 course.setName(name);  
 course.setTeacher(teacher);  
 course.setCredit(credit);  
 course.setTime(time);  
 courseRepository.save(course);  
 return true;  
}

1. Show Course:

Show the Course List which user can choose.

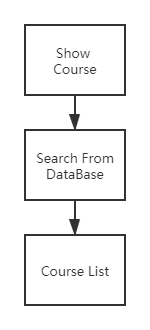
Get:

|  |  |
| --- | --- |
| Parameters | Meanings |
| None | None |

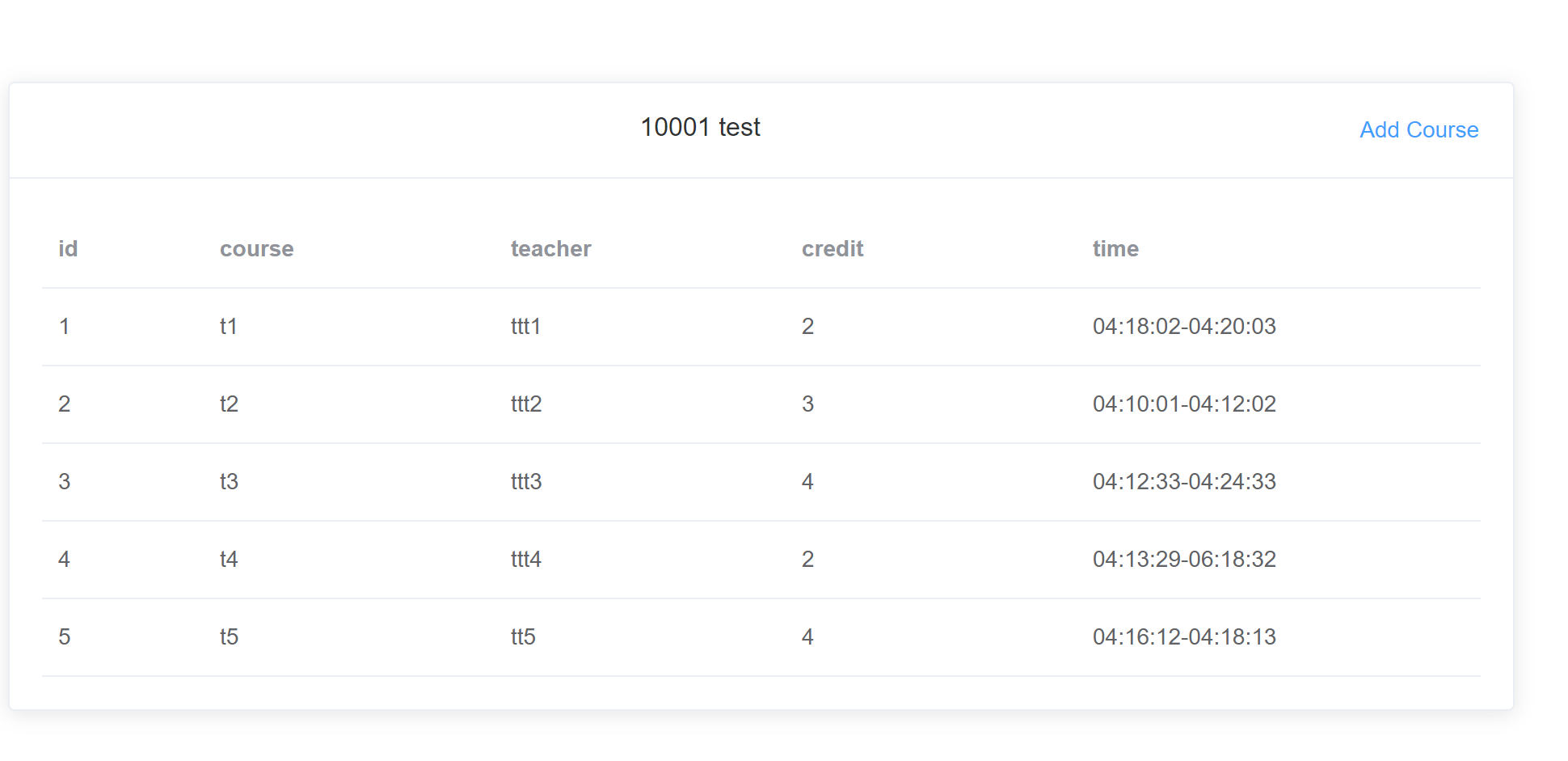
Response:

|  |  |
| --- | --- |
| Parameters | Meanings |
| Course List | The Couse List which user can choose |

Flow Chart:



Client:



Core Code:

@RequestMapping(value = "/getCourses", method = RequestMethod.GET)  
public JSONArray selectCourse() {  
 JSONArray jsonArray = new JSONArray();  
 for (Course course : courseRepository.findAll()) {  
 JSONObject json = new JSONObject();  
 json.put("id", course.getId());  
 json.put("name", course.getName());  
 json.put("teacher", course.getTeacher());  
 json.put("credit", course.getCredit());  
 json.put("time", course.getTime());  
 jsonArray.add(json);  
 }  
 return jsonArray;  
}