

Introduction

The course selection system composes of 3 components, the Vue.js client, the SparkJava/Sql2o java HTTP server and the SQL backend(sqlite3 or PostgreSQL tested-compatible).

Build Environment:

Backend

- Liberica OpenJDK 8
- Sql2o 1.6.0
- Sqlite-jdbc 3.23.3.2
- Sqlite3 3.32

Frontend

- Bootstrap CSS 5.0.0
- Vue.js v2 latest

Building

First, run clean up by “mvn clean”.

Open project root, do “mvn compile” then “mvn package”.

The provided binary automatically rollback after all SQL operations. To make permanent change, change /src/main/java/Main.java DEBUG constant to 0.

Then do a mvn cleanup and compile and package.

Automated testing

Open project root, do “mvn compile” then “mvn test”.

Test cases are in /src/test/

Running

Open /run/run.cmd with db/testsq.db under the same directory. If not found, the program would open file dialog to choose a sqlite3 db. PostgreSQL db is compatible by changing to use the commented SqlModel connection.

The server operates on port 4567. Open <http://localhost:4567/> .

	courseId	courseName	preq	weekday	start	end	cred	qta	remarks
<input type="radio"/>	cs101	IntroComput		1	1	2	2	5	collisionA
<input type="radio"/>	cs102	IntroComput2	cs101	1	1	1	1	5	collisionA
<input type="radio"/>	cs103	cs103eName		2	1	3	3	5	
<input type="radio"/>	cs104	cs104eName		3	3	5	3	5	collisionB
<input type="radio"/>	cs109	cs109eName		4	3	3	1	5	
<input checked="" type="radio"/>	te901	Art and humanity		5	5	6	2	1	quotatest
<input type="radio"/>	te910	Philosophy		1	3	6	4	5	
<input type="radio"/>	te919	Law and ethics	te910	2	5	6	2	5	
<input type="radio"/>	te928	Responsible Management		3	5	5	1	5	collisionB
<input type="radio"/>	te937	te937eName		4	5	5	1	5	

Debug section
Checked names: te901
course[0] values: { "courseId": "cs101", "courseName": "IntroComput", "preq": "", "weekday": "1", "start": "1", "end": "2", "cred": "2", "qta": "5", "remarks": "collisionA" }
cols: { "courseId", "courseName", "preq", "weekday", "start", "end", "cred", "qta", "remarks" }
server: { "success": 2, "details": "te901 not taken by 3002\nte901 dependencies fulfilled by 3002\n3002 pass credit limit check\nte901 fits with other courses selected by 3002\nte901 courses quota full 3002\nRegistration may be revoked\nAdded te901 for 3002" }

te901 not taken by 3002
te901 dependencies fulfilled by 3002
3002 pass credit limit check

Figure 1 Interface overview.

Vue.js Client

The client handles server JSON response. It is a static AJAX page, a simple analog to SPA web project while not using Vue-router and Vue-template.

The top pane consists of 3 buttons, to request JSON array of JSON objects. Vue-for script enable parsing of array into table. Table is stylized with assistance from Bootstrap 4

SQL System

```
create table Courses
(
    courseId VARCHAR(1024),
    courseName VARCHAR(1024),
    Preq VARCHAR(1024),
    Weekday VARCHAR(1024),
    Start VARCHAR(1024),
    End VARCHAR(1024),
    Cred VARCHAR(1024),
    Qta VARCHAR(1024),
```

```
Remarks VARCHAR(1024)
);

create table Users
(
    netid VARCHAR(1024),
    name VARCHAR(1024)
);

create table History
(
    RecNo INTEGER not null
        primary key autoincrement,
    NETID VARCHAR(1024)
        references Users (netid),
    courseID VARCHAR(1024)
        references Courses (courseID)
);

create table Txn
(
    txnNo INTEGER not null
        primary key autoincrement,
    netid INTEGER not null
        references Users (netid),
    courseID varchar not null
        references Courses (courseID)
);

create unique index netid_IDX
    on Users (netid);
```

Table :The DB schema

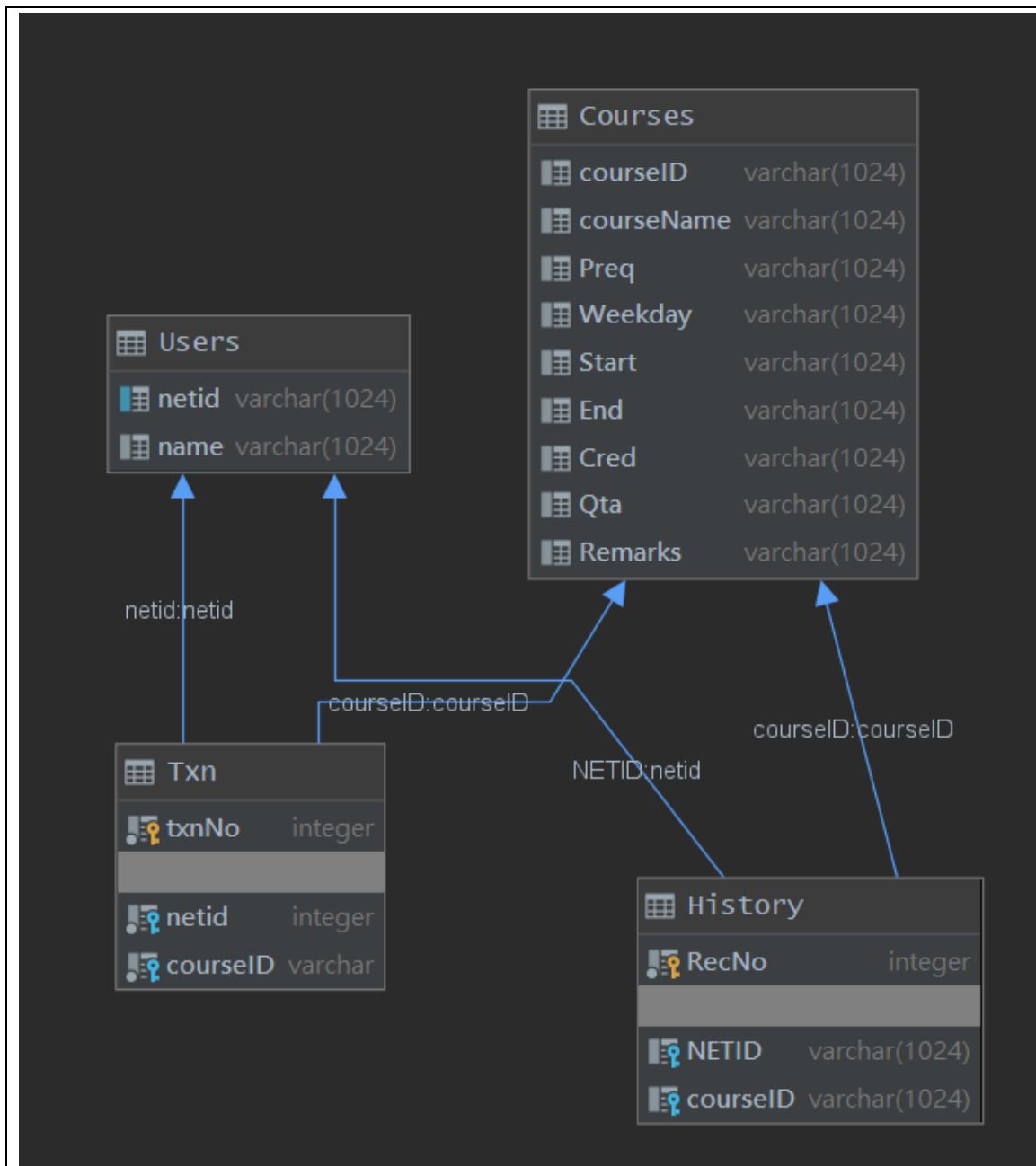


Figure 2 Entity-Relation Diagram of the database

Unit Tests for important modules

1. Drop course

Case 1: Course exist, perform delete

Case 2: Course not selected, do nothing

```
... @Test
    Run Test | Debug Test | ✓
    public void dropCourse() {
        ... SqlModel n = new SqlModel();
        ... int result1 = n.dropCourse("3001", "te901").
            getSuccess();
        ... assertThat(result1, CoreMatchers.is(1));
        ... int result2 = n.dropCourse("3001", "cs901").
            getSuccess();
        ... assertThat(result2, CoreMatchers.is(0));
        ... }
    }
```

```
SLF4J: defaulting to no-operation (NOP) logger
SLF4J: See http://www.slf4j.org/codes.html#Stat
Connection to SQLite established.
te901 chosen by 3001

before rollback
after all (rollback)
Txn [courseId=te901, netid=3001, txnNo=1]
-----
cs901 not chosen by 3001

before rollback
Txn [courseId=te901, netid=3001, txnNo=1]
after all (rollback)
Txn [courseId=te901, netid=3001, txnNo=1]
-----
```

SqlModelTest

▼ dropCourse

Passed

0.62s

Message:
N/A
Stack trace:
N/A

2. Add course

Task 1: check if taken

```
... @Test
    Run Test | Debug Test | ✓
... public void addCourse() {
...     SqlModel n = new SqlModel();
...
...     // already exist in txn
...     int result1 = n.addCourse("3001", "te901").getSuccess();
...
...     assertThat(result1, CoreMatchers.is(0));
...
...     // already taken in hist
...     int result1b = n.addCourse("3001", "cs101").getSuccess();
...
...     assertThat(result1b, CoreMatchers.is(0));
...
...
...     // not already exist
...     int result2 = n.addCourse("3009", "cs109").getSuccess();
...     assertThat(result2, CoreMatchers.is(1));
... }
```

SqlModelTest

> addCourse

Passed

0.61s

Task 2: check dependency

Example of dependency:

```
sqlite> select * from courses;
+-----+-----+-----+-----+-----+-----+-----+-----+
| courseID |   courseName   | Preq | Weekday | Start | End | Cred | Qta | Remarks |
+-----+-----+-----+-----+-----+-----+-----+-----+
| cs101    | IntroComput    |      | 1        | 1      | 2    | 2     | 5   | collisionA |
| cs102    | IntroComput2   | cs101 | 1        | 1      | 1    | 1     | 5   | collisionA |
| cs103    | cs103eName     |      | 2        | 1      | 3    | 3     | 5   |      |
| cs104    | cs104eName     |      | 3        | 3      | 5    | 3     | 5   | collisionB |
| cs109    | cs109eName     |      | 4        | 3      | 3    | 1     | 5   |      |
| te901    | Art and humanity |      | 5        | 5      | 6    | 2     | 1   | quotatest |
| te910    | Philosophy      |      | 1        | 3      | 6    | 4     | 5   |      |
| te919    | Law and ethics  | te910 | 2        | 5      | 6    | 2     | 5   |      |
| te928    | Responsible Management |      | 3        | 5      | 5    | 1     | 5   | collisionB |
| te937    | te937eName     |      | 4        | 5      | 5    | 1     | 5   |      |
+-----+-----+-----+-----+-----+-----+-----+-----+

sqlite> select preq from courses where courseid == 'cs109'
...> ;
+-----+
| Preq |
+-----+
|      |
+-----+

sqlite> select preq from courses where courseid == 'cs102'
...> ;
+-----+
| Preq |
+-----+
| cs101 |
+-----+
```

Designed test:

```
Run Test | Debug Test | ✓
... public void checkDep() {
...     SqlModel n = new SqlModel();

...     // te919 requires te910, fulfilled in txn
...     boolean result3a = n.checkDependency
...         ("te919", "3001");
...     assertThat(result3a, CoreMatchers.is
...         (true));

...     // cs102 requires cs101, fulfilled in
...     // history
...     boolean result3a2 = n.checkDependency
...         ("te919", "3001");
...     assertThat(result3a2, CoreMatchers.is
...         (true));

...     // te919 requires te910, not fulfilled
...     boolean result3b = n.checkDependency
...         ("te919", "3009");
...     assertThat(result3b, CoreMatchers.is
...         (false));

...     // te910 requires nothing
...     boolean result4 = n.checkDependency
...         ("te910", "3001");
...     assertThat(result4, CoreMatchers.is
...         (true));
... }
```

Result: Passed.

Task 3: Credit limit test

```
..@Test
  Run Test | Debug Test | ✓
..public void creditCheck() {
..    // user 3001 has selected 6
    credit
..    SqlModel n = new SqlModel();

..    boolean result1 = n.
        checkIfCreditExceed("3001",
            "20");
..    assertThat(result1,
        CoreMatchers.is(true));
..
..    boolean result2 = n.
        checkIfCreditExceed("3001",
            "3");
..    assertThat(result2,
        CoreMatchers.is(false));
..
..    boolean result3 = n.
        checkIfCreditExceed("3001",
            "15");
..    assertThat(result3,
        CoreMatchers.is(false));
..}
```

```
SLF4J: See http://www.slf4j.org/cod
Connection to SQLite established.
cred sum=26
cred sum=9
cred sum=21
```


Task 4: Collision test

```
...@Test
    Run Test | Debug Test | ✓
    ...public void decisionTest() {
        .....// c, t equal start
        .....assertTrue(decision(1, 1, 1, 1));
        .....assertTrue(decision(1, 2, 1, 2));
        .....assertTrue(decision(1, 3, 1, 2));
        .....assertTrue(decision(1, 2, 1, 3));

        .....// c, t equal end
        .....assertTrue(decision(1, 3, 2, 3));
        .....assertTrue(decision(2, 3, 1, 3));

        .....// t first
        .....assertFalse(decision(5, 6, 2, 4));
        .....assertTrue(decision(5, 6, 2, 5));

        .....// c first
        .....assertFalse(decision(2, 4, 5, 5));
        .....assertTrue(decision(2, 5, 4, 7));
    }

    .....
    .....// course cs102 collide cs101
    .....// user 1001 has cs101, try add cs102
    .....// pass dependency but fail collision
    .....SqlModel n = new SqlModel();
    .....int result1 = n.addCourse("1001", "cs102").getSuccess();
    .....assertEquals(result1, 0);
}
```

All 1 Passed 1

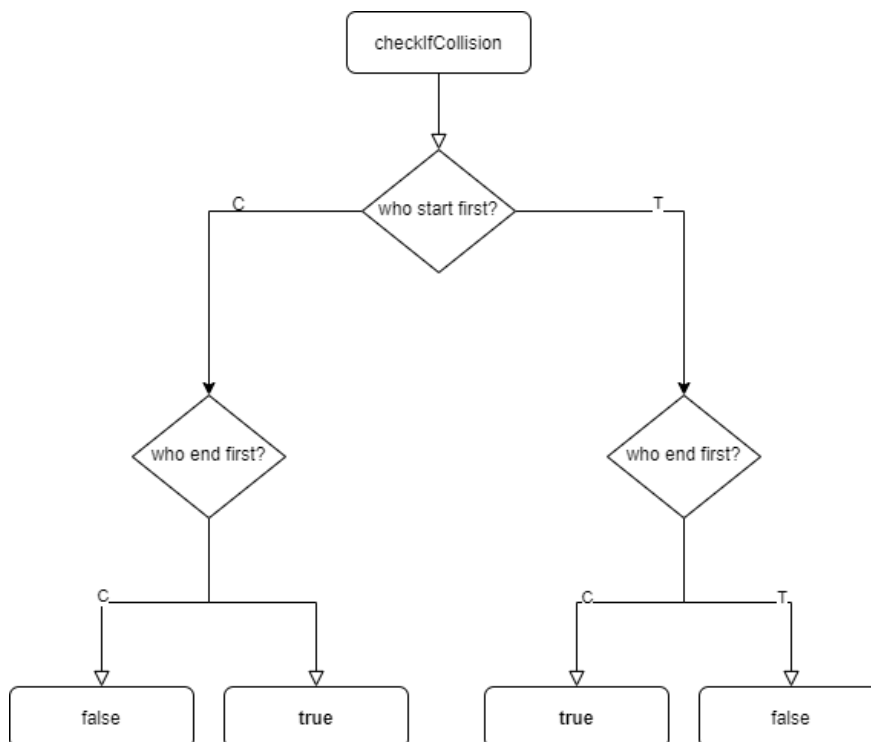
SqlModelTest

✓ decisionTest

Message:
N/A

Stack trace:
N/A

Logic:



Task 5: Quota test

(still allow register, but inform user the selection is not guaranteed)

```
...
@Test
    Run Test | Debug Test | ✓
    public void checkIfQuotaExceed() {
        ... SqlModel n = new SqlModel();
        ...
        ... //te901 is already selected by 1 person
        ... // te901 has 1 quota
        ... Courses test1 = new Courses();
        ... test1.setQta("1");
        ... test1.setCourseId("te901");
        ...
        ... boolean t = n.checkIfQuotaExceed(test1);
        ... assertTrue(t);
        ...
        ... // cs101 has 1/5 selected in txn
        ... Courses test2 = new Courses();
        ... test2.setQta("5");
        ... test2.setCourseId("cs101");
        ...
        ... boolean t2 = n.checkIfQuotaExceed(test2);
        ... assertFalse(t2);
    }
}
```

Test passed.

Task 6: insert to db for real

```

Run Test | Debug Test | ✓
... public void takenCourse() {
...     SqlModel n = new SqlModel();

...     // already exist in txn
...     int result1 = n.addCourse("3001", "te901").
...         getSuccess();

...     assertThat(result1, CoreMatchers.is(0));

...     // already taken in hist
...     int result1b = n.addCourse("3001", "cs101").
...         getSuccess();

...     assertThat(result1b, CoreMatchers.is(0));

...     // not already exist
...     int result2 = n.addCourse("3009", "cs109").
...         getSuccess();
...     assertThat(result2, CoreMatchers.is(1));
... }

```

SLF4J: See <http://www.slf4j.org/codes.html#StaticLogge>

Connection to SQLite established.

te901 already taken by 3001

cs101 already taken by 3001

cred sum=1

count +1 =2

maxQuota =5

before rollback

Txn [courseId=te901, netid=3001, txnNo=1]

Txn [courseId=te910, netid=3001, txnNo=2]

Txn [courseId=cs109, netid=3002, txnNo=3]

Txn [courseId=cs101, netid=1001, txnNo=4]

Txn [courseId=cs109, netid=3009, txnNo=5]

after all (rollback)

Txn [courseId=te901, netid=3001, txnNo=1]

Txn [courseId=te910, netid=3001, txnNo=2]

Txn [courseId=cs109, netid=3002, txnNo=3]

Txn [courseId=cs101, netid=1001, txnNo=4]

cs109 not taken by 3009

cs109 dependencies fulfilled by 3009

3009 pass credit limit check

cs109 fits with other courses selected by 3009

cs109 courses still have quota 3009

Added cs109 for 3009

Test pass.

Integral test

```
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/te910?User=3009
{
  "success" : 1,
  "details" : "te910 not taken by 3009\nte910 dependencies fulfilled by 3009\n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/te910?User=3001
{
  "success" : 0,
  "details" : "te910 already taken by 3001\n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/te910?User=3009
{
  "success" : 1,
  "details" : "te910 not taken by 3009\nte910 dependencies fulfilled by 3009\n"
}
```

```
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/cs101?User=3001
{
  "success" : 0,
  "details" : "cs101 already taken by 3001\n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/cs101?User=3002
{
  "success" : 1,
  "details" : "cs101 not taken by 3002\ncs101 dependencies fulfilled by 3002\n"
}
```

```
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/cs102?User=3001
{
  "success" : 1,
  "details" : "cs102 not taken by 3001\ncs102 dependencies fulfilled by 3001\n3001 pass credit limit check \ncs102 fits with other courses selected by 3001\n cs102 courses still have quota 3001\n"
}
```

```
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/add/te901?User=3003
{
  "success" : 2,
  "details" : "te901 not taken by 3003\nte901 dependencies fulfilled by 3003\n3003 pass credit limit check \nte901 fits with other courses selected by 3003\n te901 courses quota full 3003\nRegistration may be revoked \n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>
```

Backend http responses for add course

```

}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/
drop/te901?User=3001
{
  "success" : 1,
  "details" : "Dropped te901 chosen by 3001\n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/
drop/te910?User=3001
{
  "success" : 1,
  "details" : "Dropped te910 chosen by 3001\n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/
drop/te919?User=3001
{
  "success" : 0,
  "details" : "te919 not chosen by 3001\n"
}
C:\Users\ricts\Documents\GitHub\DBS-selectCourse\src\test>curl localhost:4567/

```

Backend http response for drop course

<input checked="" type="radio"/>	te901	Art and humanity		5	5
<input type="radio"/>	te910	Philosophy		1	3
<input type="radio"/>	te919	Law and ethics	te910	2	5
<input type="radio"/>	te928	Responsible Management		3	5
<input type="radio"/>	te937	te937eName		4	5

Debug section

Checked names: te901

course[0] values: { "courseId": "cs101", "courseName": "IntroComput", "preq": "", "weekday": "1", "start": "1", "end": "2", "cred": "2", "qta": "5", "remarks": "collisionA" }

cols: ["courseId", "courseName", "preq", "weekday", "start", "end", "cred", "qta", "remarks"]

server: { "success": 0, "details": "te901 already taken by 3001\n" }

Add Checked Courses

te937

te937eName

Debug section

Checked names: te937

course[0] values: { "courseId": "cs101", "courseName": "IntroComput", "preq": "", "weekday": "1", "start": "1", "end": "2", "cred": "3", "qta": "3", "remarks": "3002 pass credit limit check" }

cols: ["courseId", "courseName", "preq", "weekday", "start", "end", "cred", "qta", "remarks"]

server: { "success": 1, "details": "te937 not taken by 3002\nte937 dependencies fulfilled by 3002\n3002 pass credit limit check" }

```

te937 not taken by 3002
te937 dependencies fulfilled by 3002
3002 pass credit limit check
te937 fits with other courses selected by 3002
te937 courses still have quota 3002
Added te937 for 3002

```

Add Checked Courses

☒

te901

Art and humanity

☐

te910

Philosophy

☐

te919

Law and ethics

☐

te928

Responsible Management

☐

te937

te937eName

Debug section

Checked names: te901

course[0] values: { "courseId": "cs101", "courseName": "IntroComput", "preq": "", "weekday": "1", "start": "1", "end": "2", "cred": "3", "qta": "3", "remarks": "3002 pass credit limit check" }

cols: ["courseId", "courseName", "preq", "weekday", "start", "end", "cred", "qta", "remarks"]

server: { "success": 2, "details": "te901 not taken by 3002\nte901 dependencies fulfilled by 3002\n3002 pass credit limit check\nte901 fits with other courses selected by 3002\nte901 courses quota full 3002\nRegistration may be revoked\nAdded te901 for 3002" }

```

te901 not taken by 3002
te901 dependencies fulfilled by 3002
3002 pass credit limit check
te901 fits with other courses selected by 3002
te901 courses quota full 3002
Registration may be revoked
Added te901 for 3002

```

Add Checked Courses

Frontend feedback for add course(success, info: quota full)

Welcome 3001

Get courses

My courses (view/remove)

Studied Course History (view)

	txnNo	netid
<input checked="" type="radio"/>	1	3001
<input type="radio"/>	2	3001

Debug section

Checked names: te901

course[0] values: { "txnNo": 1, "netid": "3001", "courseId": "te901" }

cols: ["txnNo", "netid", "courseId"]

server: { "success": 1, "details": "Dropped te901 chosen by 3001\n" }

Remove Checked Courses

Debug section

Checked names: cs102

course[0] values: { "courseId": "cs101", "courseName": "IntroComput", "preq": "", "weekday": "1", "start": "1", "end": "2", "cred": "2", "qta": "5", "remarks": "collisionA" }

cols: ["courseId", "courseName", "preq", "weekday", "start", "end", "cred", "qta", "remarks"]

server: { "success": 0, "details": "cs102 not taken by 3002\n cs102 dependencies not fulfilled by 3002\n" }

cs102 not taken by 3002
cs102 dependencies not fulfilled by 3002

Add Checked Courses

Frontend feedback for drop course

Success Boolean and message are correctly delivered.