

Distributed System :: Final Report

1. Team member information and role

201713070 이옥희 – 33.3%

– 클라이언트 타이머 및 이벤트 설정 및 전송, 클라이언트 GUI, 전체 오류 검증 및 수정

201713074 임예린 – 33.3%

– 로그인 시 서버 타이머 설정, 타이머 관련 메시지 전송/처리, 전체 오류 검증 및 수정

201713082 주지혜 – 33.4%

- 시간 단위 동기화 메시지 전송(서버), 클라이언트 이벤트 설정, 전체 오류 검증 및 수정

2. Project goal

(option 1) To develop an application using CM

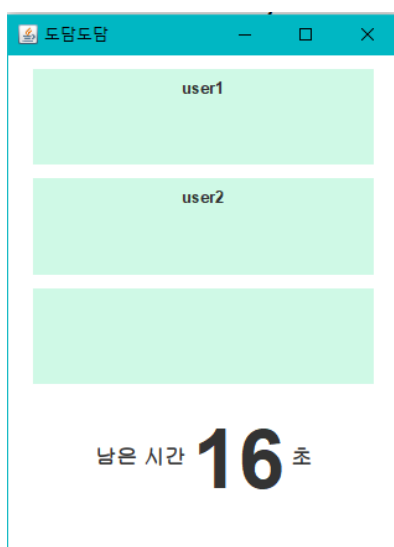
도담도담 – 공부로 키우는 나무

도담도담은 사용자가 공부에 집중할 수 있도록 도움을 주는 응용프로그램. 임의의 사용자들이 그룹을 이루어 주어진 시간 동안 나무를 키운다. 사용자의 이벤트를 감지하여 주어진 시간 내에 마우스를 움직이거나 키보드 입력이 있을 경우 그룹 내에서 자라고 있던 나무가 죽는다.

3. Project design

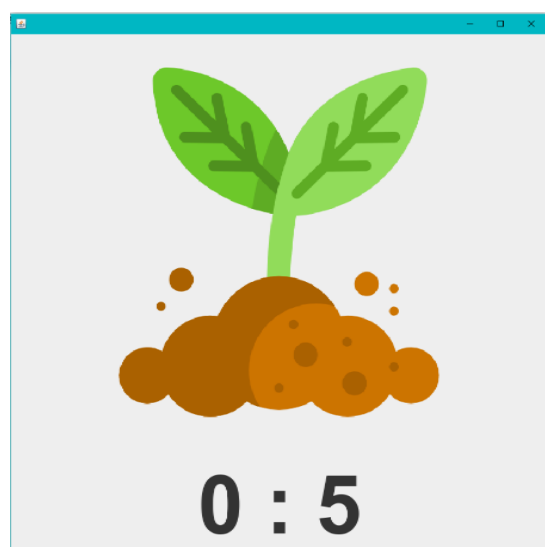
A. 도담도담 응용프로그램의 실행 화면

A-1)



사용자 그룹 생성 화면

A-2)



공부 중 나무 상태 & 타이머

A-3) 도담도담 프로토콜

CMDummyEvent로 메시지 전송. 구분자는 세미콜론(;)이다.

1. 서버가 보내는 메시지

LOGIN;
SECOND;

*LOGIN : 대기 타이머 정보

START;
TREE_STATE

*START (프로그램 실행 정보)

- START : 시작 OR 실행 중
- END : 사용자 이벤트가 발생한 경우 종료 OR 정상 종료

*TREE_STATE (나무 상태 정보)

- TREE_SEED : 초기 상태
- TREE_1H : 1시간 지난 상태
- TREE_2H : 2시간 지난 상태
- TREE_SUCCESS : 3시간 지난 상태 (정상 종료)
- TREE_DEAD : 사용자 이벤트 발생으로 죽은 상태 (종료)

2. 클라이언트가 보내는 메시지

TIME

* TIME (실행 중 클라이언트의 타이머 정보)

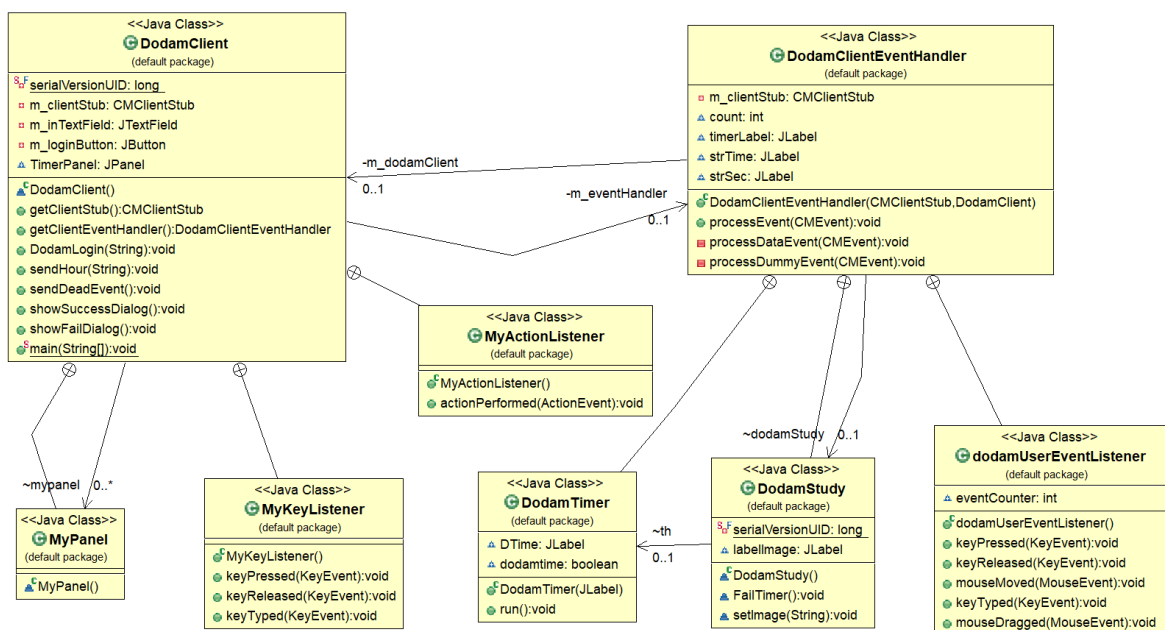
- ONE_HOUR : 1시간 지남
- TWO_HOUR : 2시간 지남
- THREE_HOUR : 3시간 지남 (정상 종료)

DEAD

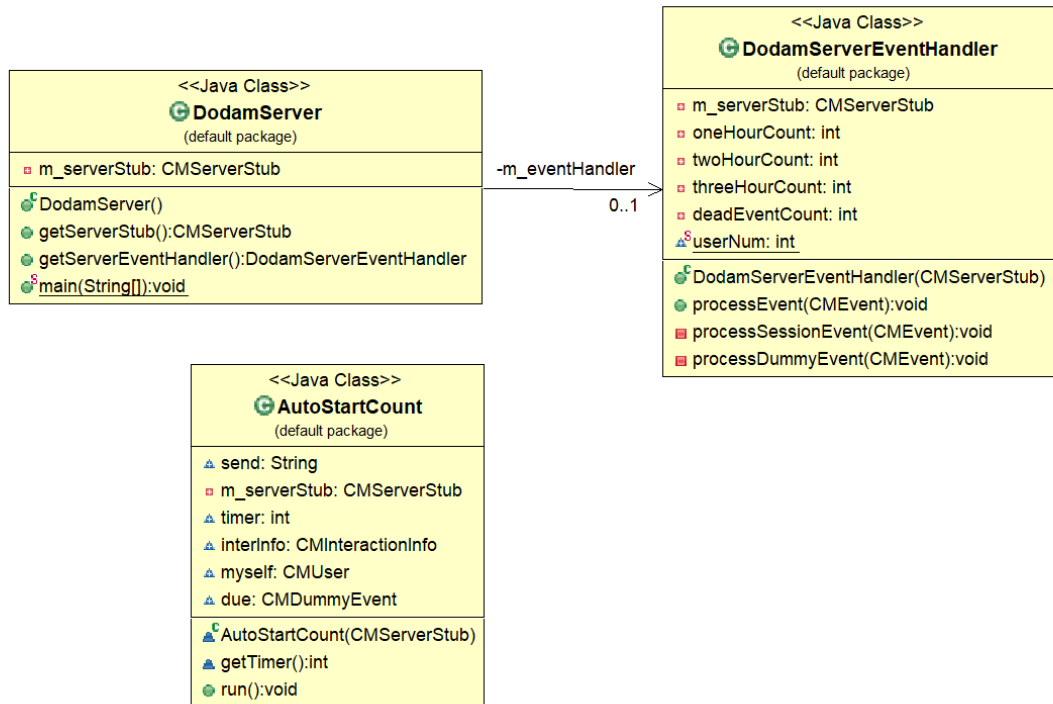
* DEAD : 사용자 이벤트로 종료

A-4) 클래스 다이어그램

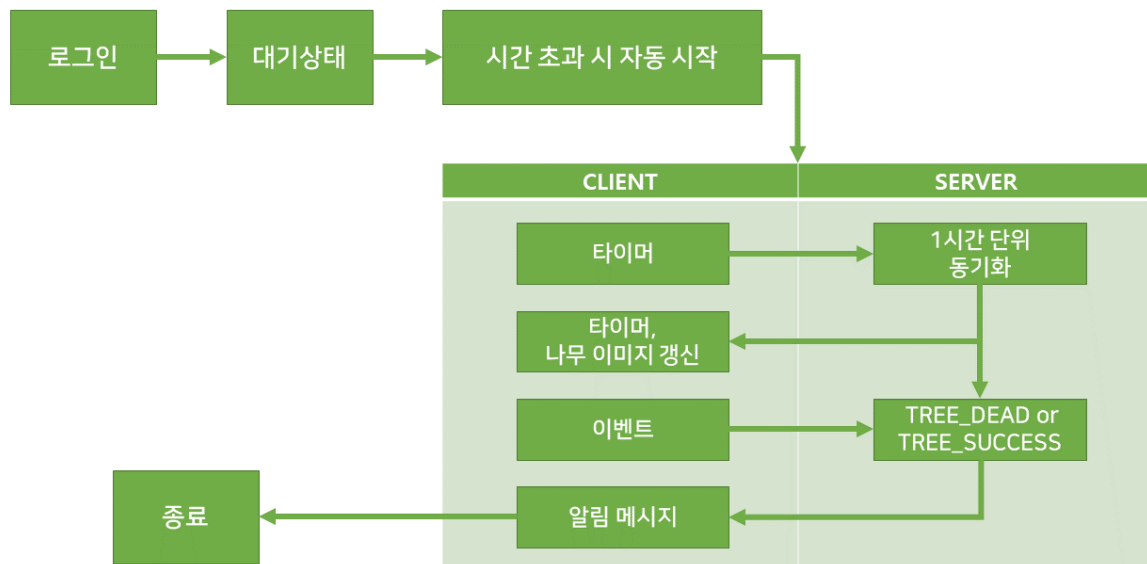
1. 클라이언트



2. 서버



A-5) 실행 플로우차트



로그인 : 사용자로부터 ID를 입력받는다. ID를 입력한 사용자는 별도의 확인 절차 없이 로그인에 성공한다.

대기상태 : 그림 A-1) 화면. 사용자는 이 화면에서 자동시작 될 때까지 대기한다.

이 상태에서, 서버는 클라이언트에게 남은 대기시간을 전송한다.

자동시작 : 그림 A-1)의 타이머가 종료되었을 경우 자동으로 A-2)의 화면으로 넘어간다.

A-2) 화면으로 바뀌고 시작이 되면

1- 클라이언트는 타이머를 실행하고, 1시간 단위로 서버에게 메시지를 보낸다.

A. 클라이언트 타이머 정보

ONE_HOUR : 1시간 지났을 때 클라이언트가 보내는 메시지

TWO_HOUR : 2시간 지났을 때 클라이언트가 보내는 메시지

THREE_HOUR : 3시간 지났을 때(정상 종료 시) 클라이언트가 보내는 메시지

2- 서버는 클라이언트의 메시지를 받아 현재 그룹 안에 있는 모든 클라이언트들에게 트리 상태 정보를 보내 싱크를 맞춘다.

3- 클라이언트는 서버가 보내 준 메시지를 토큰으로 분리하고, 분석한다.

4- 분석한 메시지에 따라 나무 이미지를 갱신한다.

A. 트리 상태 정보 **테스트의 편의를 위해 1시간을 10초로 치환한다.*

TREE_SEED : 처음 시작 상태의 나무. 씨앗 그림

TREE_1H : 1시간이 지난 상태의 나무. 새싹 그림

TREE_2H : 2시간이 지난 상태의 나무. 작은 나무 그림

TREE_SUCCESS : 3시간이 지나 다 자란 나무. 큰 나무 그림

TREE_DEAD : 사용자가 3시간이 지나기 전 마우스나 키보드 등을 사용해 실패했을 경우의 죽은 나무. 썩은 나무 그림

5- 이벤트 발생 및 종료

A. 성공적으로 3시간이 지났을 경우 (사용자 이벤트가 발생하지 않음)

i. 서버는 클라이언트에게 TREE_SUCCESS 신호를 보내고 알림 메시지를 띄운 후 종료한다.

B. 중도에 사용자 이벤트(마우스 혹은 키보드)가 발생했을 경우

i. TREE_DEAD 신호를 보내고 실패 메시지를 띄운 후 종료한다.

B. How to use CM

- **synchronous login API in the CM client stub.**

public CMSessionEvent syncLoginCM(String strUserName, String strPassword)

: 클라이언트에서 로그인할 때 사용

- **send method**

boolean send(CMEvent cme, String strTarget)

: 클라이언트에서 타이머 이벤트 또는 사용자 이벤트가 발생한 경우 서버로 CMDummyEvent를 전송할 때 사용 (아래의 두 사용자 정의 함수에서 사용됨)

*public void sendHour(String hour) *사용자 정의 함수*

: 1시간이 지날 때마다 클라이언트가 서버에게 시간 정보를 담은 CMDummyEvent 전송

*public void sendDeadEvent() *사용자 정의 함수*

: 사용자 이벤트(마우스 움직임 또는 키 입력)가 발생한 경우 클라이언트가 서버에게 종료를 알리는 CMDummyEvent 전송

- **cast method**

boolean cast(CMEvent cme, String sessionName, String groupName)

: 클라이언트에서 사용자가 로그인 시 서버에서 대기 시간 타이머 값을 보내는 경우 또는 클라이언트로부터 받은 이벤트를 처리한 결과를 보내는 경우에 사용

**클라이언트에서는 서버로만 CMDummyEvent를 보내기 때문에 send method를 사용하고,*

서버는 session1, group1에 있는 모든 클라이언트에게 CMDummyEvent를 보내기 때문에 서버에서는 cast method를 사용한다.

- **CM_DATA_EVENT**

processDataEvent(cme);

: 사용자가 로그인 했을 때 클라이언트의 사용자 목록을 갱신한다.

(1) CMDataEvent.NEW_USER

: 새로운 사용자를 포함한 기존의 사용자들에게 새로 들어온 사용자의 ID를 알려준다. 새 사용자의 ID를 받은 다른 사용자들은 사용자 목록에 ID를 추가한다.

(2) CMDataEvent.INHABITANT

: 기존의 사용자들이 새로 들어온 사용자에게 자신의 ID를 알려준다. 기존 사용자들의 ID를 받은 새로운 사용자는 사용자 목록에 ID를 추가한다.

- **CM_SESSION_EVENT**

CMSessionEvent.LOGIN

: 서버에서 로그인할 때 사용. 로그인 시 사용자를 세션 내의 그룹에 넣고 관리하기 위해 사용. 로그인 시 호출된다.

- **CM_DUMMY_EVENT**

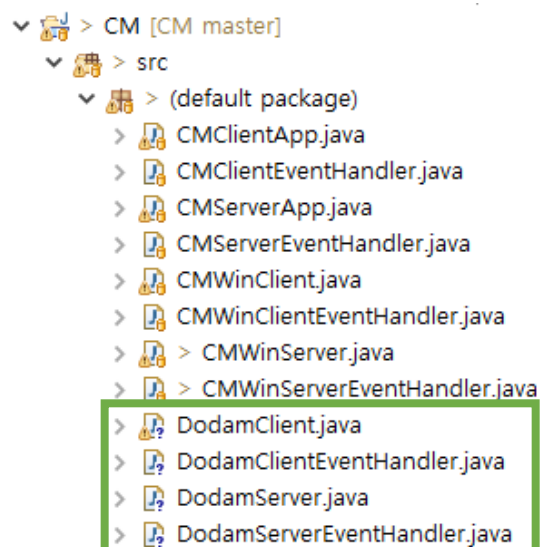
processDummyEvent(cme);

: 서버와 클라이언트 간 메시지 전송 시 사용 (로그인, 타이머 전송, 시간 별 메시지 전송)

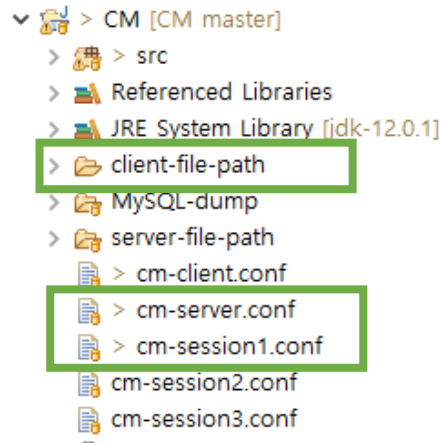
4. Project result

B. Describe how to run and operate your system.

- CM 프로그램을 설치한다.
- CM 프로그램 내부의 Default package에 도담도담 코드 파일(DodamServer.java, DodamServerEventHandler.java, DodamClient.java, DodamClientEventHandler.java)을 넣는다.



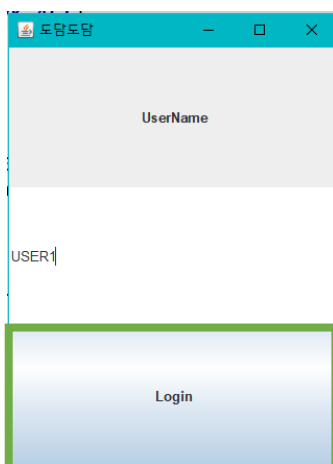
- iii. 기존 CM의 conf파일 중 cm-server.conf, cm-session1.conf를 도담도담 코드 파일과 함께 업로드한 conf 파일들로, 기존 client-file-path를 이미지 파일들이 들어 있는 client-file-path로 덮어씌운다.



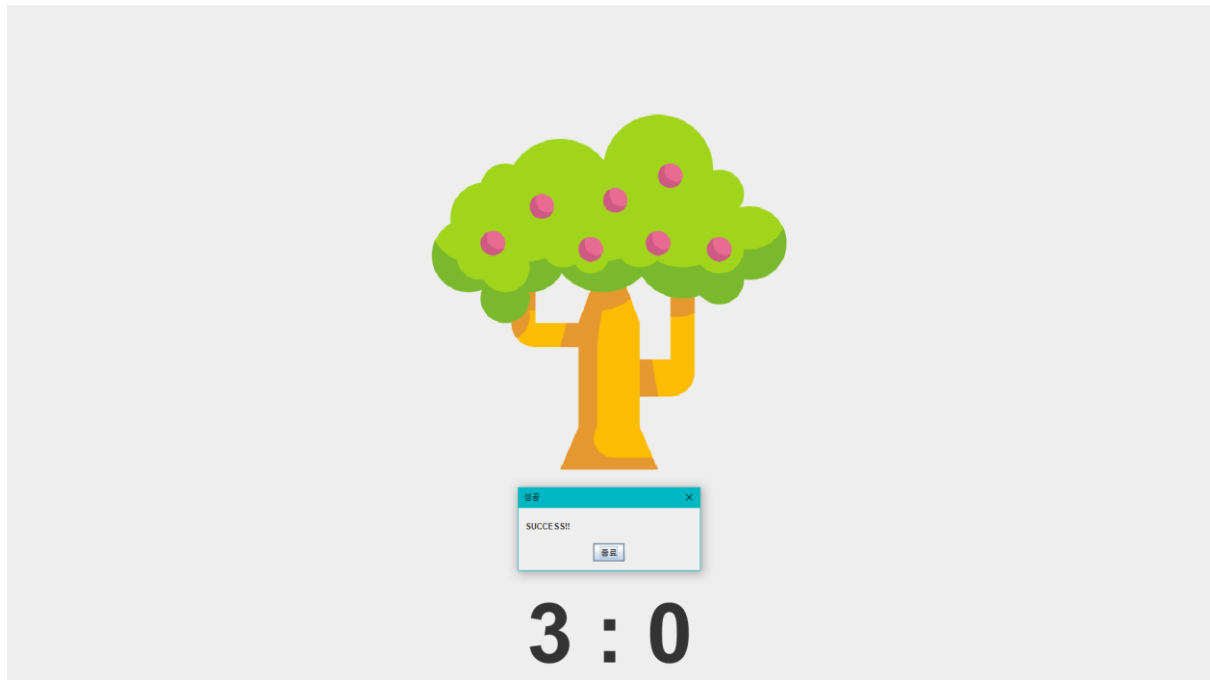
- iv. cm-server.conf와 cm-client.conf 파일들의 ip 주소를 지정하고, cm-server.conf의 LOGIN_SCHEME, SESSION_SCHEME, DB_USE를 0으로 설정한다.

```
45 LOGIN_SCHEME 0
46
47 # session scheme (true / false)
48 # true: use multiple sessions and region
49 # false: no session and region (use a de
50 SESSION_SCHEME 0
51
52 ##### CM DB configuration
53
54 # DB usage (true/ false)
55 # true: use DB (MySQL) internally for us
56 # false: do not use DB
57 DB_USE 0
```

- v. DodamServer.java를 실행한 후, DodamClient.java를 실행한다.
- vi. [DodamClient] 임의의 문자열을 입력한 후 로그인 버튼을 누르면 로그인된다. 자동시작타이머 30초가 지나면 공부 중 나무 상태와 타이머 화면이 나타난다.



- vii. [DodamClient] 마우스나 키보드 입력 없이 3시간(편의 1시간을 10초로 잡아 총 30초로 설정되어 있음)이 지나면 나무는 열매를 맺고 성공적으로 종료한다.



- viii. [DodamClient] 3시간(임의 30초)이 지나기 전에 마우스나 키보드 입력이 있을 경우 죽은 나무의 이미지를 출력하고 종료한다.

