

StudySmart

CS157A Project Final Report

Kyle Wang, Minh Phong Do, Joshua Demo, Nathan Wong

Team 4

Table of Contents

| | |
|-----------------------------|----|
| Project Requirements..... | 2 |
| Project Design..... | 6 |
| Project Implementation..... | 20 |
| Lessons Learned..... | 54 |

Project Requirements

StudySmart is a student-focused application that allows users to manage academic and extracurricular activities. The user will be able to add, update, and delete sections for their notes, homework, and outside activities. Tasks are automatically prioritized according to the deadline and are filtered by type or status. The application also includes a progress tracker that will record study hours, task completion, and grades, and an attendance tracking to keep track of missing lectures. StudySmart is a detailed tool for students to remain organized and productive while keeping study habits healthy.

Functions:

User Information:

- Register:

Users can create an account by providing their name, email, and password, with email verification for security.

- Setup Profile:

After registering, users can customize their profile by adding courses and extracurricular activities for a personalized experience.

- Login:

Users can log in securely with their email and password to access the app's features. The system will validate credentials.

- Update Profile:

Users can look at all their information and change their name, password, email, and profile picture. The change password has a confirmation to make sure the entered password is what the user wants.

Task Management and Workstation Features

- MyTask Page Features:
 - Users can create new tasks with comprehensive details including:
 - Task name and description
 - Deadline
 - Priority levels (high, medium, low)
 - Status tracking (pending, complete, overdue)
 - Task status automatically updates based on:
 - User-marked completion status
 - Current date in relation to the deadline
 - Organization features include:
 - Sorting by deadline
 - Sorting by priority levels
 - Filtering options based on task status
- Workstation Page Features
 - Accessed by clicking on any specific task
 - Task modification capabilities include:
 - Updating task descriptions

- Adding and managing notes
- Attaching various resources
- Updating completion status
- Linking tasks to registered courses
- Comprehensive information display:
 - Shows all task-related details
 - Displays deadline information
 - Presents attached resources and notes
- Study Mode
 - Recording functionality:
 - Start/stop recording of study sessions to track how long the user has been studying
 - Access to recorded sessions through the study sessions page
 - Enhanced study environment:
 - Integrated music playback feature
 - Ability to study while managing tasks
- Resource Management
 - Support for various online resource types:
 - Video links
 - Google Docs
 - Images
 - Homework website links
 - Other online educational resources

Progress Tracking:

- Monitor Progress:

Users can track their academic progress through the Progress Page, where they can:

- Add assignments to update grades for their courses.
- Track their total study hours to see how much time they've spent on each subject or task.

- Manage Courses

- Users can add or delete courses
- Each course has its own section for assignments
- You can add or delete specific assignments within each course

- Manage Grades:

- Users can add or update grades for their courses, which are linked to specific assignments within each course.

- Manage Extracurriculars:

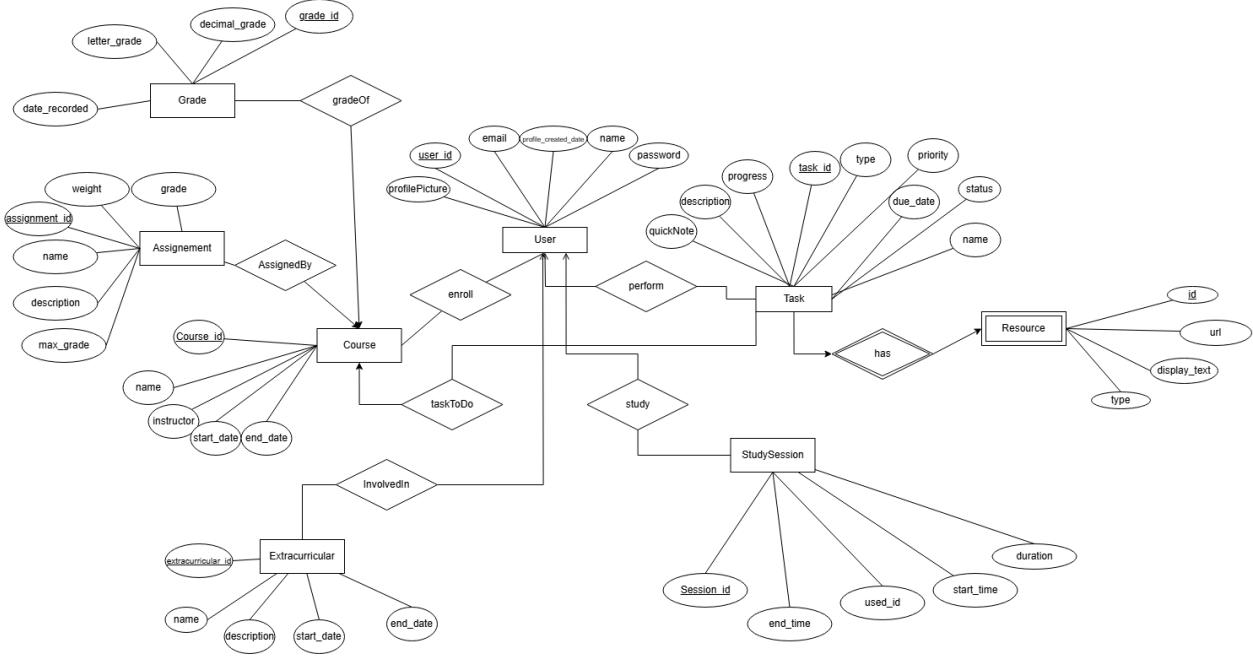
- Users can add extracurriculars with values such as the description, date, and name

- View Task Completion:

A task completeness bar helps users monitor how much of their workload is complete, showing their progress at a glance.

Project Design

Entity-Relationship Diagram

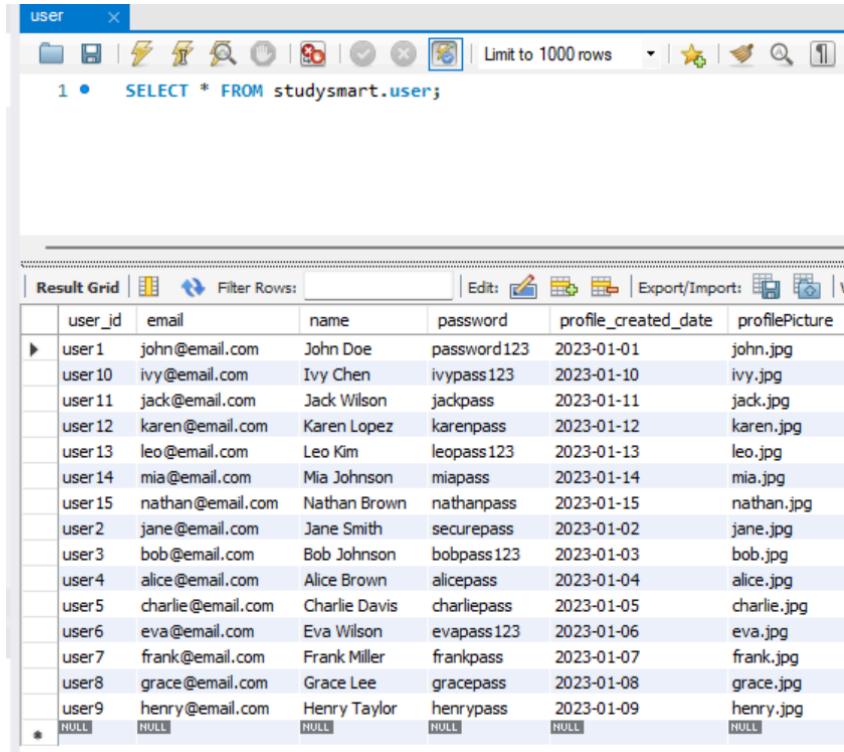


Entities

User: Represents a student using the StudySmart application.

- user_id (pk)
- email
- name
- password
- profile_created_date
- profilePicture

User Schema: User(user_id, email, name, password, profile_created_date, profilePicture)



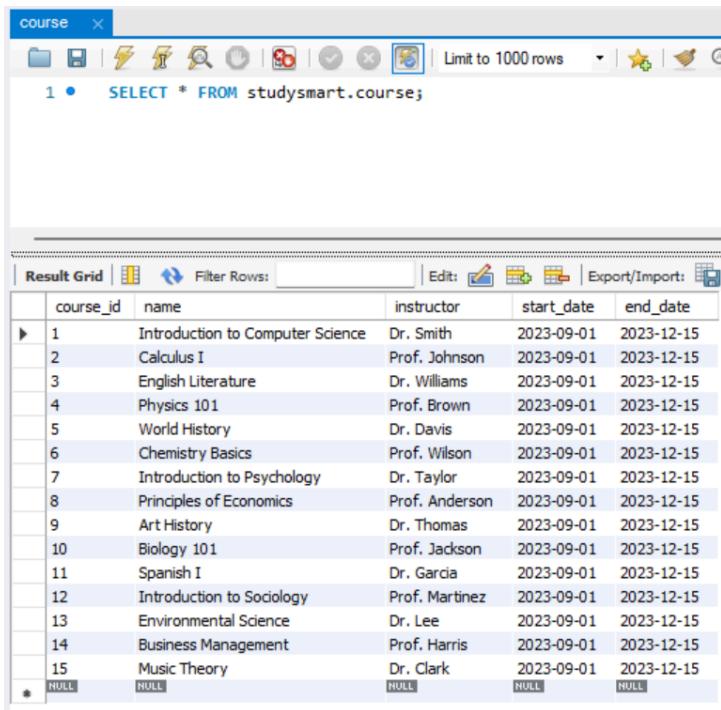
The screenshot shows the MySQL Workbench interface with a query editor window titled 'user'. The query is 'SELECT * FROM studysmart.user;'. Below the query, the 'Result Grid' displays 15 rows of data from the 'user' table. The columns are: user_id, email, name, password, profile_created_date, and profilePicture. The data includes various user entries with their corresponding details and profile pictures.

| | user_id | email | name | password | profile_created_date | profilePicture |
|---|---------|-------------------|---------------|-------------|----------------------|----------------|
| ▶ | user1 | john@email.com | John Doe | password123 | 2023-01-01 | john.jpg |
| | user10 | ivy@email.com | Ivy Chen | ivypass123 | 2023-01-10 | ivy.jpg |
| | user11 | jack@email.com | Jack Wilson | jackpass | 2023-01-11 | jack.jpg |
| | user12 | karen@email.com | Karen Lopez | karenpass | 2023-01-12 | karen.jpg |
| | user13 | leo@email.com | Leo Kim | leopass123 | 2023-01-13 | leo.jpg |
| | user14 | mia@email.com | Mia Johnson | miapass | 2023-01-14 | mia.jpg |
| | user15 | nathan@email.com | Nathan Brown | nathanpass | 2023-01-15 | nathan.jpg |
| | user2 | jane@email.com | Jane Smith | securepass | 2023-01-02 | jane.jpg |
| | user3 | bob@email.com | Bob Johnson | bobpass123 | 2023-01-03 | bob.jpg |
| | user4 | alice@email.com | Alice Brown | alicepass | 2023-01-04 | alice.jpg |
| | user5 | charlie@email.com | Charlie Davis | charliepass | 2023-01-05 | charlie.jpg |
| | user6 | eva@email.com | Eva Wilson | evapass123 | 2023-01-06 | eva.jpg |
| | user7 | frank@email.com | Frank Miller | frankpass | 2023-01-07 | frank.jpg |
| | user8 | grace@email.com | Grace Lee | gracepass | 2023-01-08 | grace.jpg |
| | user9 | henry@email.com | Henry Taylor | henrypass | 2023-01-09 | henry.jpg |
| * | HULL | HULL | HULL | HULL | HULL | HULL |

Course: Represents an academic course.

- course_id (pk)
- name
- instructor
- start_date
- end_date

Course Schema: Course(course_id, name, instructor, start_date, end_date)



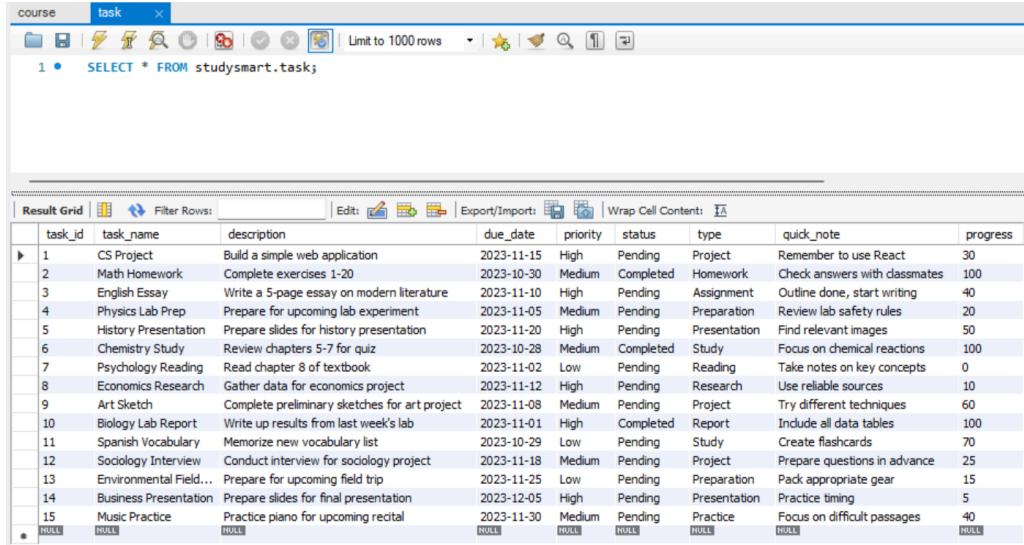
The screenshot shows the MySQL Workbench interface with a query editor window titled 'course'. The query 'SELECT * FROM studysmart.course;' is entered. Below the query, the results are shown in a 'Result Grid'.

| course_id | name | instructor | start_date | end_date |
|-----------|----------------------------------|----------------|------------|------------|
| 1 | Introduction to Computer Science | Dr. Smith | 2023-09-01 | 2023-12-15 |
| 2 | Calculus I | Prof. Johnson | 2023-09-01 | 2023-12-15 |
| 3 | English Literature | Dr. Williams | 2023-09-01 | 2023-12-15 |
| 4 | Physics 101 | Prof. Brown | 2023-09-01 | 2023-12-15 |
| 5 | World History | Dr. Davis | 2023-09-01 | 2023-12-15 |
| 6 | Chemistry Basics | Prof. Wilson | 2023-09-01 | 2023-12-15 |
| 7 | Introduction to Psychology | Dr. Taylor | 2023-09-01 | 2023-12-15 |
| 8 | Principles of Economics | Prof. Anderson | 2023-09-01 | 2023-12-15 |
| 9 | Art History | Dr. Thomas | 2023-09-01 | 2023-12-15 |
| 10 | Biology 101 | Prof. Jackson | 2023-09-01 | 2023-12-15 |
| 11 | Spanish I | Dr. Garcia | 2023-09-01 | 2023-12-15 |
| 12 | Introduction to Sociology | Prof. Martinez | 2023-09-01 | 2023-12-15 |
| 13 | Environmental Science | Dr. Lee | 2023-09-01 | 2023-12-15 |
| 14 | Business Management | Prof. Harris | 2023-09-01 | 2023-12-15 |
| 15 | Music Theory | Dr. Clark | 2023-09-01 | 2023-12-15 |
| * | NULL | NULL | NULL | NULL |

Task: Represents any task a user needs to complete.

- task_id (pk)
- task_name
- description
- due_date
- priority (enum: Low, Medium, High)
- status (enum: Pending, Completed, Overdue)
- type
- quick_note
- Progress

Task Schema: Task(task_id, task_name, description, due_date, priority, status, type, quick_note, progress)



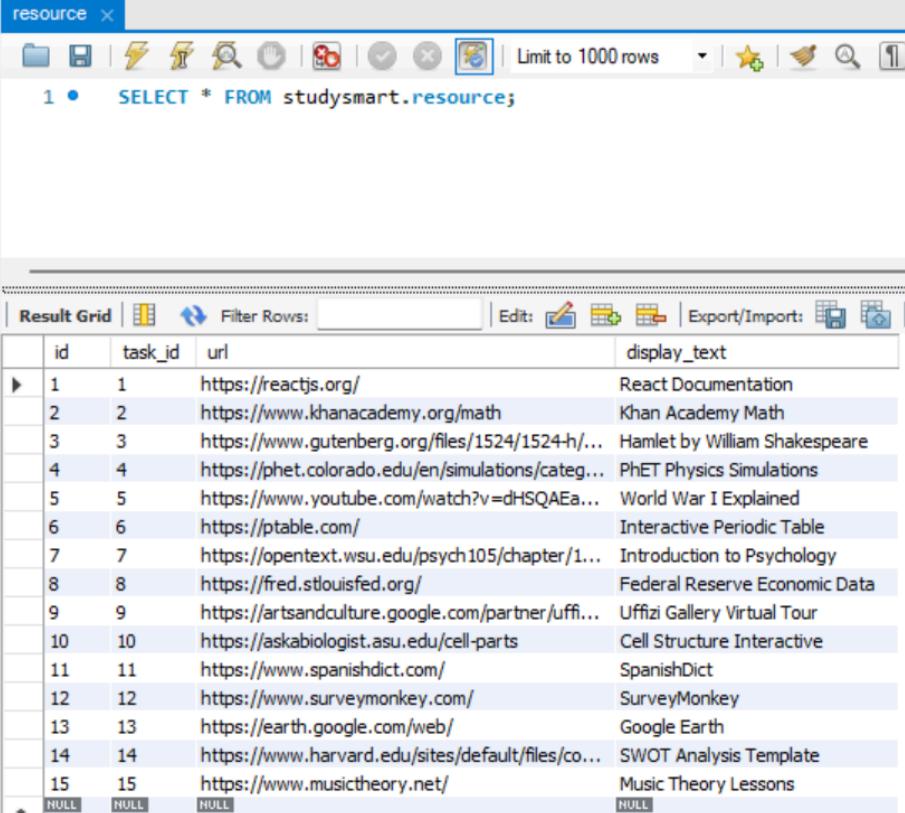
The screenshot shows a database interface with a toolbar at the top containing various icons for file operations, search, and export. Below the toolbar, a SQL query is displayed: "1 • SELECT * FROM studysmart.task;". The main area is a "Result Grid" showing the data from the query. The grid has columns: task_id, task_name, description, due_date, priority, status, type, quick_note, and progress. There are 15 rows of data, each representing a task with its details.

| | task_id | task_name | description | due_date | priority | status | type | quick_note | progress |
|---|---------|------------------------|---|------------|----------|-----------|--------------|-------------------------------|----------|
| ▶ | 1 | CS Project | Build a simple web application | 2023-11-15 | High | Pending | Project | Remember to use React | 30 |
| | 2 | Math Homework | Complete exercises 1-20 | 2023-10-30 | Medium | Completed | Homework | Check answers with classmates | 100 |
| | 3 | English Essay | Write a 5-page essay on modern literature | 2023-11-10 | High | Pending | Assignment | Outline done, start writing | 40 |
| | 4 | Physics Lab Prep | Prepare for upcoming lab experiment | 2023-11-05 | Medium | Pending | Preparation | Review lab safety rules | 20 |
| | 5 | History Presentation | Prepare slides for history presentation | 2023-11-20 | High | Pending | Presentation | Find relevant images | 50 |
| | 6 | Chemistry Study | Review chapters 5-7 for quiz | 2023-10-28 | Medium | Completed | Study | Focus on chemical reactions | 100 |
| | 7 | Psychology Reading | Read chapter 8 of textbook | 2023-11-02 | Low | Pending | Reading | Take notes on key concepts | 0 |
| | 8 | Economics Research | Gather data for economics project | 2023-11-12 | High | Pending | Research | Use reliable sources | 10 |
| | 9 | Art Sketch | Complete preliminary sketches for art project | 2023-11-08 | Medium | Pending | Project | Try different techniques | 60 |
| | 10 | Biology Lab Report | Write up results from last week's lab | 2023-11-01 | High | Completed | Report | Include all data tables | 100 |
| | 11 | Spanish Vocabulary | Memorize new vocabulary list | 2023-10-29 | Low | Pending | Study | Create flashcards | 70 |
| | 12 | Sociology Interview | Conduct interview for sociology project | 2023-11-18 | Medium | Pending | Project | Prepare questions in advance | 25 |
| | 13 | Environmental Field... | Prepare for upcoming field trip | 2023-11-25 | Low | Pending | Preparation | Pack appropriate gear | 15 |
| | 14 | Business Presentation | Prepare slides for final presentation | 2023-12-05 | High | Pending | Presentation | Practice timing | 5 |
| | 15 | Music Practice | Practice piano for upcoming recital | 2023-11-30 | Medium | Pending | Practice | Focus on difficult passages | 40 |
| * | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

Resource: Represents resources linked to specific tasks.

- id (pk)
- task_id (fk)
- url
- display_text
- type

Resource Schema: Resource(id, task_id, url, display_text, type)



The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query editor contains the SQL command: `SELECT * FROM studysmart.resource;`. The results grid displays 15 rows of data from the 'resource' table, with columns: id, task_id, url, and display_text.

| | id | task_id | url | display_text |
|---|------|---------|---|-------------------------------|
| ▶ | 1 | 1 | https://reactjs.org/ | React Documentation |
| | 2 | 2 | https://www.khanacademy.org/math | Khan Academy Math |
| | 3 | 3 | https://www.gutenberg.org/files/1524/1524-h/... | Hamlet by William Shakespeare |
| | 4 | 4 | https://phet.colorado.edu/en/simulations/categ... | PhET Physics Simulations |
| | 5 | 5 | https://www.youtube.com/watch?v=dHSQAEa... | World War I Explained |
| | 6 | 6 | https://ptable.com/ | Interactive Periodic Table |
| | 7 | 7 | https://opentext.wsu.edu/psych105/chapter/1... | Introduction to Psychology |
| | 8 | 8 | https://fred.stlouisfed.org/ | Federal Reserve Economic Data |
| | 9 | 9 | https://artsandculture.google.com/partner/uffi... | Uffizi Gallery Virtual Tour |
| | 10 | 10 | https://askabiologist.asu.edu/cell-parts | Cell Structure Interactive |
| | 11 | 11 | https://www.spanishdict.com/ | SpanishDict |
| | 12 | 12 | https://www.surveymonkey.com/ | SurveyMonkey |
| | 13 | 13 | https://earth.google.com/web/ | Google Earth |
| | 14 | 14 | https://www.harvard.edu/sites/default/files/co... | SWOT Analysis Template |
| | 15 | 15 | https://www.musictheory.net/ | Music Theory Lessons |
| * | NULL | NULL | NULL | NULL |

StudySession: Tracks study sessions.

- session_id (pk)
- start_time
- end_time
- date_recorded

StudySession Schema: StudySession(session_id, start_time, end_time, date_recorded)

```

    SELECT * FROM studysmart.studySession;
  
```

| | session_id | start_time | end_time | date_recorded |
|----|------------|------------|------------|---------------|
| 1 | 09:00:00 | 11:00:00 | 2023-10-15 | |
| 2 | 14:00:00 | 16:00:00 | 2023-10-16 | |
| 3 | 10:00:00 | 12:00:00 | 2023-10-17 | |
| 4 | 13:00:00 | 15:00:00 | 2023-10-18 | |
| 5 | 11:00:00 | 13:00:00 | 2023-10-19 | |
| 6 | 15:00:00 | 17:00:00 | 2023-10-20 | |
| 7 | 09:30:00 | 11:30:00 | 2023-10-21 | |
| 8 | 14:30:00 | 16:30:00 | 2023-10-22 | |
| 9 | 10:30:00 | 12:30:00 | 2023-10-23 | |
| 10 | 13:30:00 | 15:30:00 | 2023-10-24 | |
| 11 | 11:30:00 | 13:30:00 | 2023-10-25 | |
| 12 | 15:30:00 | 17:30:00 | 2023-10-26 | |
| 13 | 09:00:00 | 11:00:00 | 2023-10-27 | |
| 14 | 14:00:00 | 16:00:00 | 2023-10-28 | |
| 15 | 10:00:00 | 12:00:00 | 2023-10-29 | |
| * | NULL | NULL | NULL | |

Grade: Represents grades.

- grade_id (pk)
- decimal_grade
- letter_grade
- date_recorded

Grade Schema: Grade(grade_id, decimal_grade, letter_grade, date_recorded)

| | grade_id | decimal_grade | letter_grade | date_recorded |
|----|----------|---------------|--------------|---------------|
| 1 | 95.5 | A | 2023-10-15 | |
| 2 | 99.2 | A | 2023-10-16 | |
| 3 | 97.6 | A | 2023-10-17 | |
| 4 | 96.7 | A | 2023-10-18 | |
| 5 | 98.75 | A | 2023-10-19 | |
| 6 | 98.2 | A | 2023-10-20 | |
| 7 | 98 | A | 2023-10-21 | |
| 8 | 96.1 | A | 2023-10-22 | |
| 9 | 98.6 | A | 2023-10-23 | |
| 10 | 97.75 | A | 2023-10-24 | |
| 11 | 96 | A | 2023-10-25 | |
| 12 | 98 | A | 2023-10-26 | |
| 13 | 97.6 | A | 2023-10-27 | |
| 14 | 95.8 | A | 2023-10-28 | |
| 15 | 87.75 | B | 2023-10-29 | |
| * | NULL | NULL | NULL | |

Extracurricular: Represents non-academic activities.

- extracurricular_id (pk)
- activity_name
- description
- start_date
- end_date

Extracurricular Schema: Extracurricular(extracurricular_id, activity_name, description, start_date, end_date)

```
1 •   SELECT * FROM studysmart.extracurricular;
```

| | extracurricular_id | activity_name | description | start_date | end_date |
|---|--------------------|----------------------|--|------------|------------|
| ▶ | 1 | Chess Club | Weekly chess meetings and tournaments | 2023-09-01 | 2023-12-15 |
| ▶ | 2 | Debate Team | Participate in inter-college debate competitions | 2023-09-01 | 2023-12-15 |
| ▶ | 3 | Volunteer Group | Community service activities | 2023-09-01 | 2023-12-15 |
| ▶ | 4 | Soccer Team | College soccer team practices and matches | 2023-09-01 | 2023-12-15 |
| ▶ | 5 | Art Club | Weekly art workshops and exhibitions | 2023-09-01 | 2023-12-15 |
| ▶ | 6 | Coding Club | Collaborative coding projects and hackathons | 2023-09-01 | 2023-12-15 |
| ▶ | 7 | Environmental Club | Campus sustainability initiatives | 2023-09-01 | 2023-12-15 |
| ▶ | 8 | Theater Group | Rehearsals and performances of plays | 2023-09-01 | 2023-12-15 |
| ▶ | 9 | Music Band | Practice sessions and campus performances | 2023-09-01 | 2023-12-15 |
| ▶ | 10 | Photography Club | Photo walks and exhibition preparations | 2023-09-01 | 2023-12-15 |
| ▶ | 11 | Robotics Team | Building and programming robots for competit... | 2023-09-01 | 2023-12-15 |
| ▶ | 12 | Language Exchange | Weekly meetings for practicing foreign langua... | 2023-09-01 | 2023-12-15 |
| ▶ | 13 | Fitness Club | Group workouts and fitness challenges | 2023-09-01 | 2023-12-15 |
| ▶ | 14 | Entrepreneurship ... | Workshops and startup pitch competitions | 2023-09-01 | 2023-12-15 |
| ▶ | 15 | Film Club | Movie screenings and discussions | 2023-09-01 | 2023-12-15 |
| * | HULL | HULL | HULL | HULL | HULL |

Assignment: Represents specific coursework assignments.

- assignment_id (pk)
- name
- description
- grade
- max_grade
- weight

Assignment Schema: Assignment(assignment_id, name, description, grade, max_grade, weight)

The screenshot shows a database interface with the following details:

- Schemas:** A tree view showing various tables like assignedby, assignment, attendance, course, enrolled, extracurricular, grade, gradeof, involvedin, note, performs, project, resource, studies, studysession, task, tasktodo, user.
- Views and Stored Procedures:** A section listing Views and Stored Procedures.
- Administration:** A section for managing the database.
- Information:** A section for viewing system information.
- Schema:** The current schema selected is **studysmart**.
- Table:** The **assignment** table is selected, indicated by the title bar and the SQL query in the editor.
- SQL Editor:** The query `SELECT * FROM studysmart.assignment;` is displayed.
- Result Grid:** The table data is presented in a grid format with the following columns: assignment_id, name, description, grade, max_grade, and weight.
- Data:** The result grid contains 15 rows of assignment data.

| assignment_id | name | description | grade | max_grade | weight |
|---------------|---------------------------|------------------------------------|-------|-----------|--------|
| 1 | CS Midterm | Midterm exam for Intro to CS | 85 | 100 | 0.3 |
| 2 | Calculus Problem Set 1 | First problem set for Calculus I | 92 | 100 | 0.1 |
| 3 | English Essay | Analysis of Shakespeare's Hamlet | 88 | 100 | 0.2 |
| 4 | Physics Lab Report | Report on Newton's Laws experiment | 78 | 100 | 0.15 |
| 5 | History Research Paper | Research on World War II | 95 | 100 | 0.25 |
| 6 | Chemistry Quiz | Quiz on periodic table | 82 | 100 | 0.1 |
| 7 | Psychology Case Study | Analysis of a psychological case | 90 | 100 | 0.2 |
| 8 | Economics Project | Market analysis project | 87 | 100 | 0.3 |
| 9 | Art Presentation | Presentation on Renaissance art | 93 | 100 | 0.2 |
| 10 | Biology Lab Report | Report on cell structure | 85 | 100 | 0.15 |
| 11 | Spanish Oral Exam | Oral examination for Spanish I | 80 | 100 | 0.2 |
| 12 | Sociology Survey Project | Design and conduct a social survey | 92 | 100 | 0.25 |
| 13 | Environmental Science ... | Report on local ecosystem | 88 | 100 | 0.2 |
| 14 | Business Case Analysis | Analysis of a business case study | 86 | 100 | 0.3 |
| 15 | Music Composition | Compose a short musical piece | 51 | 100 | 0.25 |
| * | NULL | NULL | NULL | NULL | NULL |

Relationships

1. User-Course (**Enrolled**)
 - Many-to-Many relationship through Enrolled table
 - Attributes: user_id (fk), course_id (fk)
 - **Schema:** Enrolled(user_id, course_id)

Navigator

enrolled

SCHEMAS

Filter objects

assignment
attendance
course
enrolled
extracurricular
grade
gradeof
involvedin
note
performs
project
resource
studies
studysession
task
tasktodo
user

Views
Stored Procedures
Functions

Administration Schemas

Information

Schema: studysmart

Result Grid

| | user_id | course_id |
|---|---------|-----------|
| ▶ | user1 | 1 |
| | user2 | 2 |
| | user3 | 3 |
| | user4 | 4 |
| | user5 | 5 |
| | user6 | 6 |
| | user7 | 7 |
| | user8 | 8 |
| | user9 | 9 |
| | user10 | 10 |
| | user11 | 11 |
| | user12 | 12 |
| | user13 | 13 |
| | user14 | 14 |
| | user15 | 15 |
| * | HULL | HULL |

2. User-Task (Performs)

- Many-to-Many relationship through Performs table
- Attributes: user_id (fk), task_id (fk)
- **Schema:** Performs(user_id, task_id)

performs

Filter Rows: _____

1 • SELECT * FROM studysmart.performs;

Result Grid

| | user_id | task_id |
|---|---------|---------|
| ▶ | user1 | 1 |
| | user2 | 2 |
| | user3 | 3 |
| | user4 | 4 |
| | user5 | 5 |
| | user6 | 6 |
| | user7 | 7 |
| | user8 | 8 |
| | user9 | 9 |
| | user10 | 10 |
| | user11 | 11 |
| | user12 | 12 |
| | user13 | 13 |
| | user14 | 14 |
| | user15 | 15 |
| * | HULL | HULL |

3. User-StudySession (**Studies**)

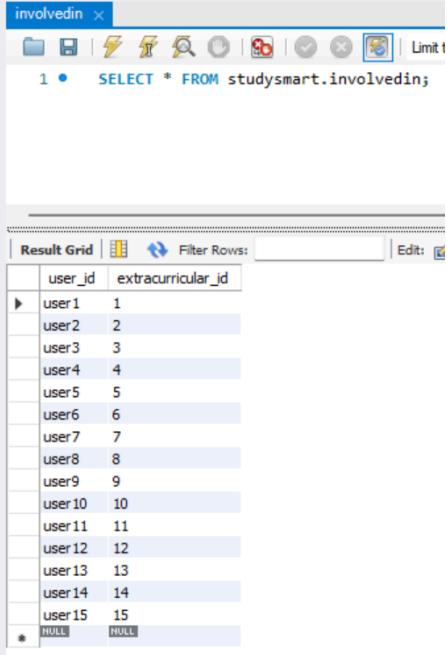
- Many-to-Many relationship through Studies table
- Attributes: user_id (fk), session_id (fk)
- **Schema:** Studies(user_id, session_id)

The screenshot shows the MySQL Workbench interface with a query editor window titled 'studies'. The query is: `SELECT * FROM studysmart.studies;`. Below the query, the result grid displays the data from the 'studies' table. The table has two columns: 'user_id' and 'session_id'. The data consists of 15 rows, each containing a user ID and a session ID. The first 14 rows have user IDs ranging from 1 to 15, and session IDs ranging from 1 to 15. The last row is a NULL entry.

| | user_id | session_id |
|---|---------|------------|
| ▶ | user1 | 1 |
| | user2 | 2 |
| | user3 | 3 |
| | user4 | 4 |
| | user5 | 5 |
| | user6 | 6 |
| | user7 | 7 |
| | user8 | 8 |
| | user9 | 9 |
| | user10 | 10 |
| | user11 | 11 |
| | user12 | 12 |
| | user13 | 13 |
| | user14 | 14 |
| | user15 | 15 |
| * | HULL | HULL |

4. User-Extracurricular (**InvolvedIn**)

- Many-to-Many relationship through Involvedin table
- Attributes: user_id (fk), extracurricular_id (fk)
- **Schema:** InvolvedIn(user_id, extracurricular_id)



The screenshot shows the MySQL Workbench interface with a query editor titled 'involvedin'. The query is:

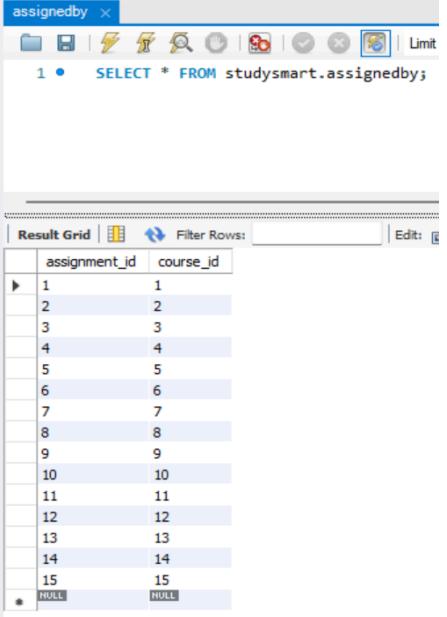
```
1 •   SELECT * FROM studysmart.involvedin;
```

The result grid displays the following data:

| | user_id | extracurricular_id |
|---|---------|--------------------|
| ▶ | user1 | 1 |
| | user2 | 2 |
| | user3 | 3 |
| | user4 | 4 |
| | user5 | 5 |
| | user6 | 6 |
| | user7 | 7 |
| | user8 | 8 |
| | user9 | 9 |
| | user10 | 10 |
| | user11 | 11 |
| | user12 | 12 |
| | user13 | 13 |
| | user14 | 14 |
| | user15 | 15 |
| * | NULL | NULL |

5. Course-Assignment (**AssignedBy**)

- Many-to-Many relationship through Assignedby table
- Attributes: assignment_id (fk), course_id (fk)
- **Schema:** AssignedBy(assignment_id, course_id)



The screenshot shows the MySQL Workbench interface with a query editor titled 'assignedby'. The query is:

```
1 •   SELECT * FROM studysmart.assignedby;
```

The result grid displays the following data:

| | assignment_id | course_id |
|---|---------------|-----------|
| ▶ | 1 | 1 |
| | 2 | 2 |
| | 3 | 3 |
| | 4 | 4 |
| | 5 | 5 |
| | 6 | 6 |
| | 7 | 7 |
| | 8 | 8 |
| | 9 | 9 |
| | 10 | 10 |
| | 11 | 11 |
| | 12 | 12 |
| | 13 | 13 |
| | 14 | 14 |
| | 15 | 15 |
| * | NULL | NULL |

6. Course-Grade (**GradeOf**)

- Many-to-Many relationship through Gradeof table
- Attributes: grade_id (fk), course_id (fk)
- **Schema:** GradeOf(grade_id, course_id)

The screenshot shows the MySQL Workbench interface with a database named 'studysmart'. A table named 'gradeof' is selected. The SQL editor at the top contains the query: `SELECT * FROM studysmart.gradeof;`. Below it, the results grid displays the following data:

| | grade_id | course_id |
|---|----------|-----------|
| ▶ | 1 | 1 |
| | 2 | 2 |
| | 3 | 3 |
| | 4 | 4 |
| | 5 | 5 |
| | 6 | 6 |
| | 7 | 7 |
| | 8 | 8 |
| | 9 | 9 |
| | 10 | 10 |
| | 11 | 11 |
| | 12 | 12 |
| | 13 | 13 |
| | 14 | 14 |
| | 15 | 15 |
| * | NULL | NULL |

7. Course-Task (**TaskToDo**)

- Many-to-Many relationship through Tasktodo table
- Attributes: course_id (fk), task_id (fk)
- **Schema:** TaskToDo(course_id, task_id)

The screenshot shows a MySQL Workbench interface with a tab titled 'tasktodo'. Below the tabs are standard database management icons. A query window displays the following SQL command:

```
1 •  SELECT * FROM studysmart.tasktodo;
```

The results are shown in a 'Result Grid' table:

| | course_id | task_id |
|---|-----------|---------|
| ▶ | 1 | 1 |
| | 2 | 2 |
| | 3 | 3 |
| | 4 | 4 |
| | 5 | 5 |
| | 6 | 6 |
| | 7 | 7 |
| | 8 | 8 |
| | 9 | 9 |
| | 10 | 10 |
| | 11 | 11 |
| | 12 | 12 |
| | 13 | 13 |
| | 14 | 14 |
| | 15 | 15 |
| * | NULL | NULL |

Weak Entity Sets

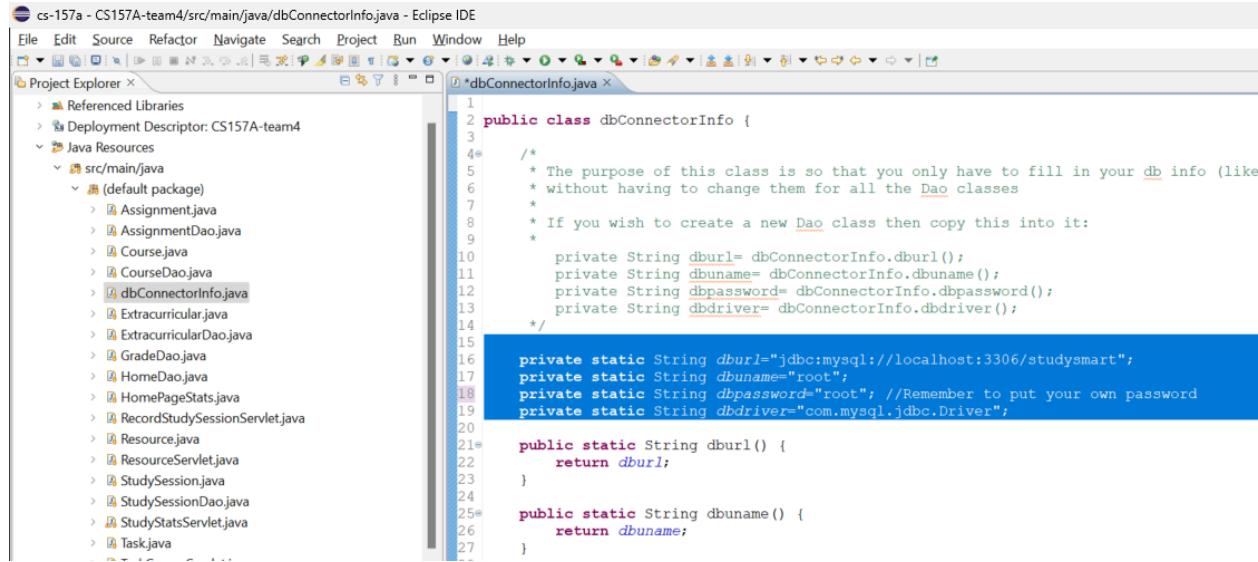
Resource is a weak entity set dependent on task

Project Implementation

Get started:

Before running the app with Tomcat server,

Make sure to change the “dbuname” and “dbpassword” in *dbConnectorInfo.java* under *src/main/java* directory to match the name and password of MySQL local instance

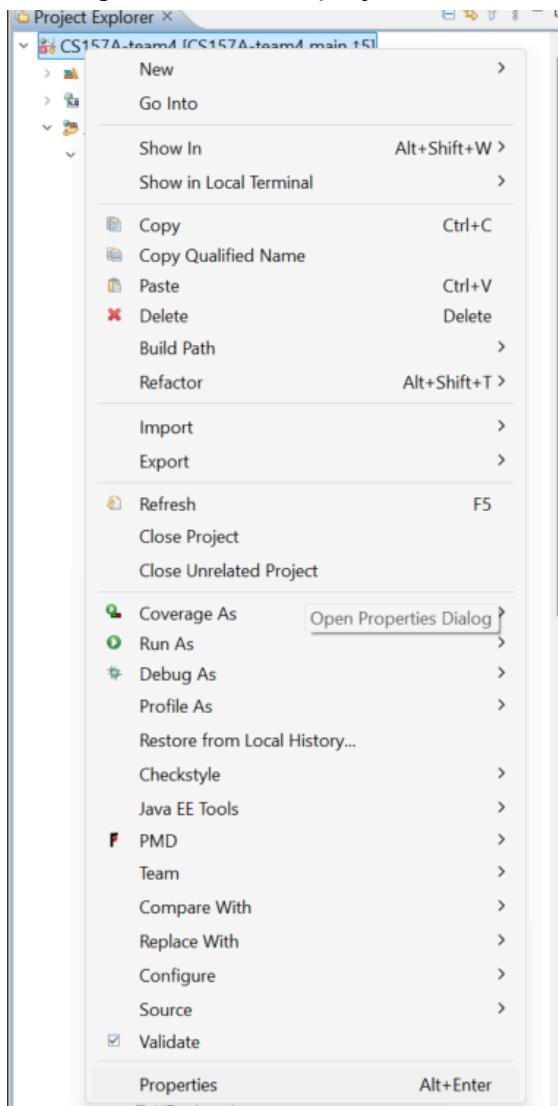


```

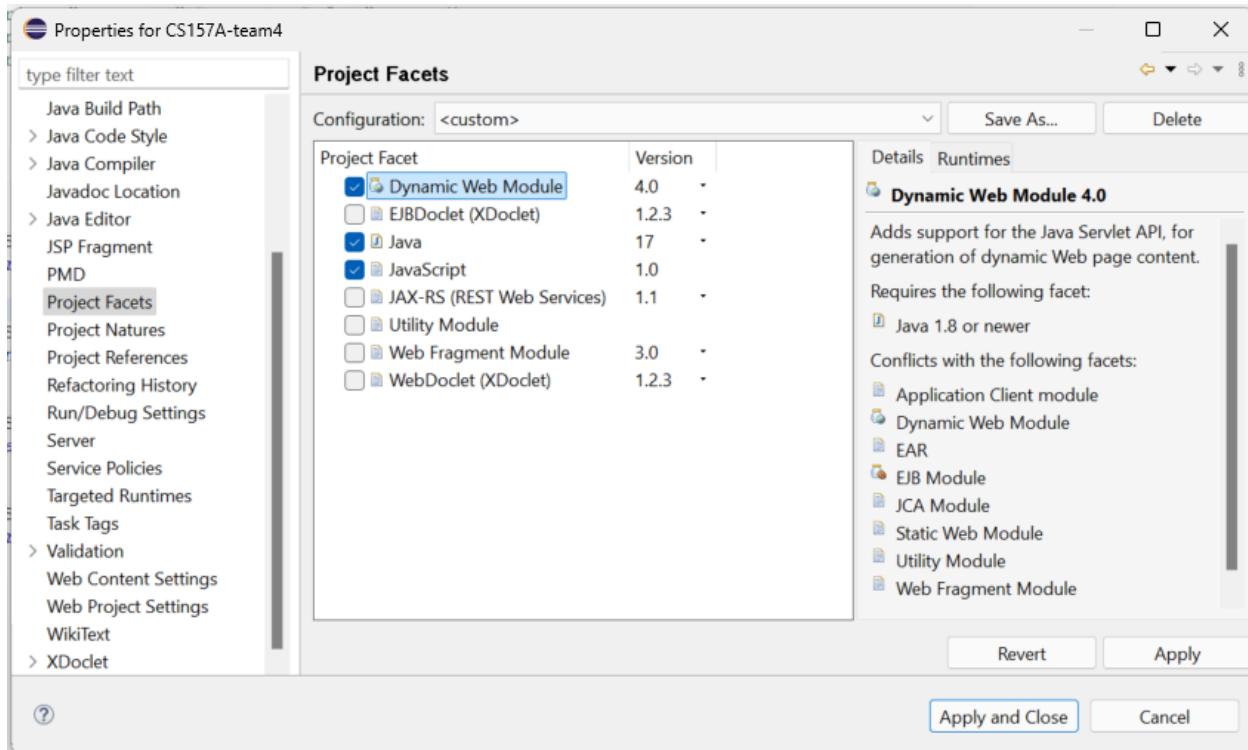
1  public class dbConnectorInfo {
2
3     /*
4      * The purpose of this class is so that you only have to fill in your db info (like
5      * without having to change them for all the Dao classes
6      *
7      * If you wish to create a new Dao class then copy this into it:
8      */
9
10    private String dburl= dbConnectorInfo.dburl();
11    private String dbuname= dbConnectorInfo.dbuname();
12    private String dbpassword= dbConnectorInfo.dbpassword();
13    private String dbdriver= dbConnectorInfo.dbdriver();
14
15
16    private static String dburl="jdbc:mysql://localhost:3306/studysmart";
17    private static String dbuname="root";
18    private static String dbpassword="root"; //Remember to put your own password
19    private static String dbdriver="com.mysql.jdbc.Driver";
20
21    public static String dburl() {
22        return dburl;
23    }
24
25    public static String dbuname() {
26        return dbuname;
27    }

```

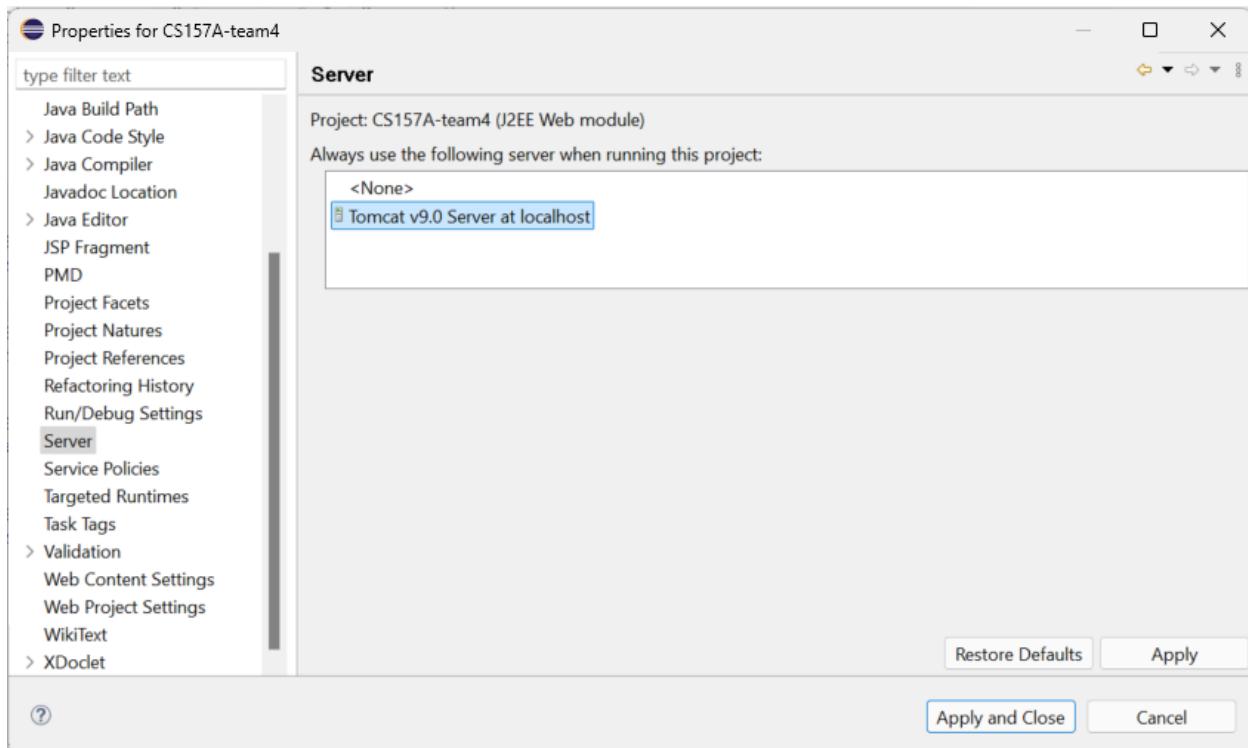
Next right click on the project folder and locate “Properties” option.



Make sure that under *Project Facets*, the version of *Dynamic Web Module*, *Java*, *JavaScript* are the same as in the picture below



Also, make sure the project is using v9.0 of Tomcat Server



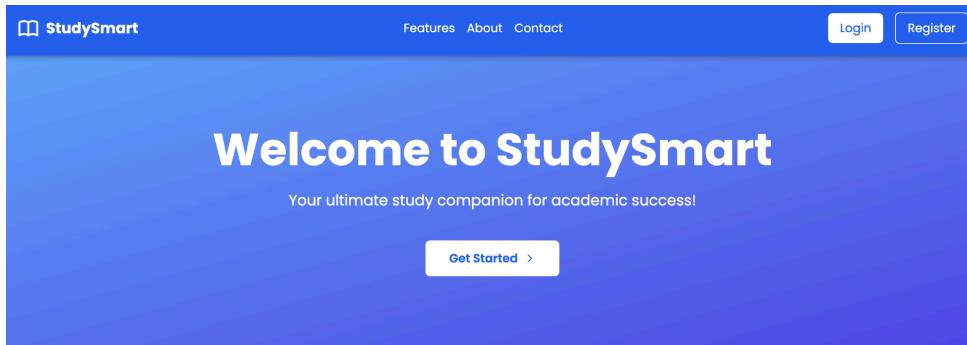
Sample account for testing:

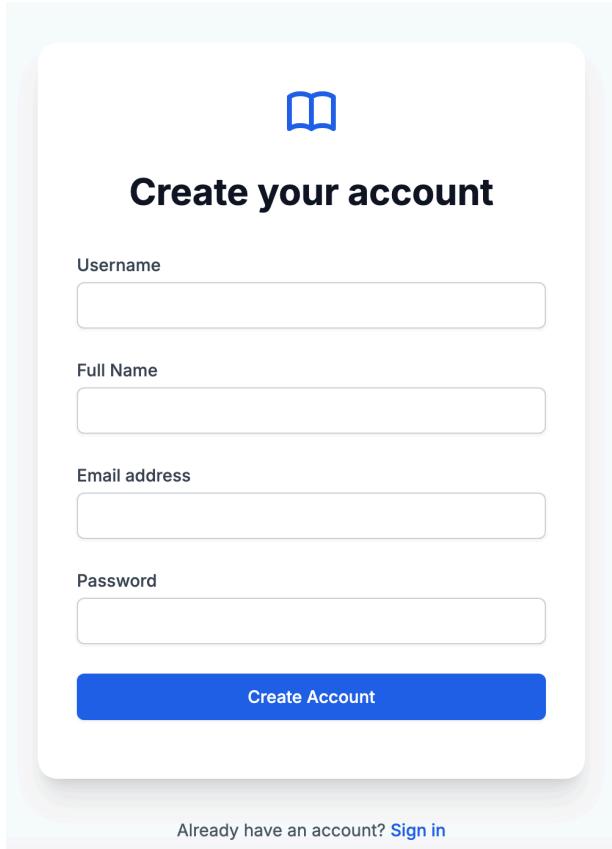
Username: domipo

Password: 123

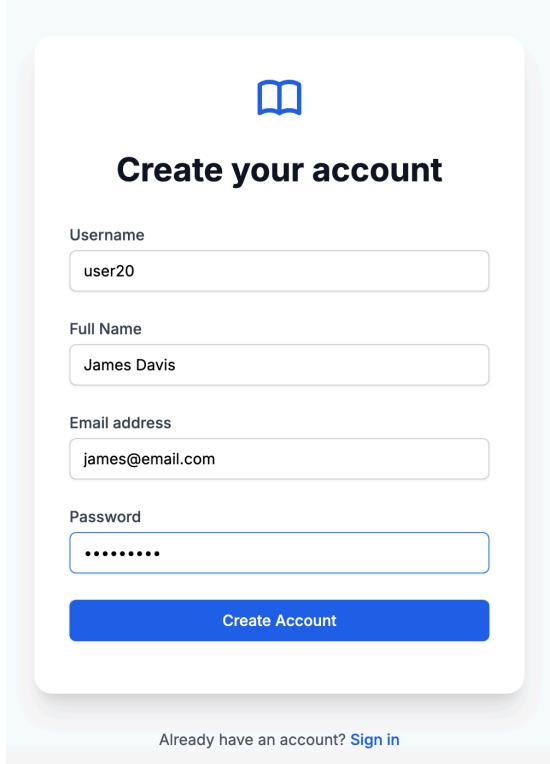
Home page:

When you first load the page up, you are taken to this page to either login or register, you can also look at features, the about, or contact the StudySmart team

**Features****Register page:**



Register takes you to the registration page, to register, input your information and press “Create Account”



If the account created is valid, this will redirect you to the homepage with your information. The nav bar is also updated to indicate the user is logged in and changing the right side to show profile and logout buttons instead of login and register buttons. Also the middle of the nav bar shows the different sections the user can go to instead of what the application is about

The dashboard features a top navigation bar with 'StudySmart' logo, 'Courses', 'Tasks', 'Study Sessions', 'Extracurriculars', 'Profile', and 'Logout'. Below the header, a welcome message 'Welcome back, user20!' is displayed. The main area contains six cards: 'My Profile' (user icon), 'My Courses' (book icon), 'Study Sessions' (clock icon), 'Extracurriculars' (puzzle piece icon), 'My Tasks' (list icon), and 'My Stats' (chart icon). The stats section shows 0 courses enrolled, 0 upcoming tasks, and 12.5 study hours this week. At the bottom, a copyright notice reads '© 2024 StudySmart. All rights reserved.'

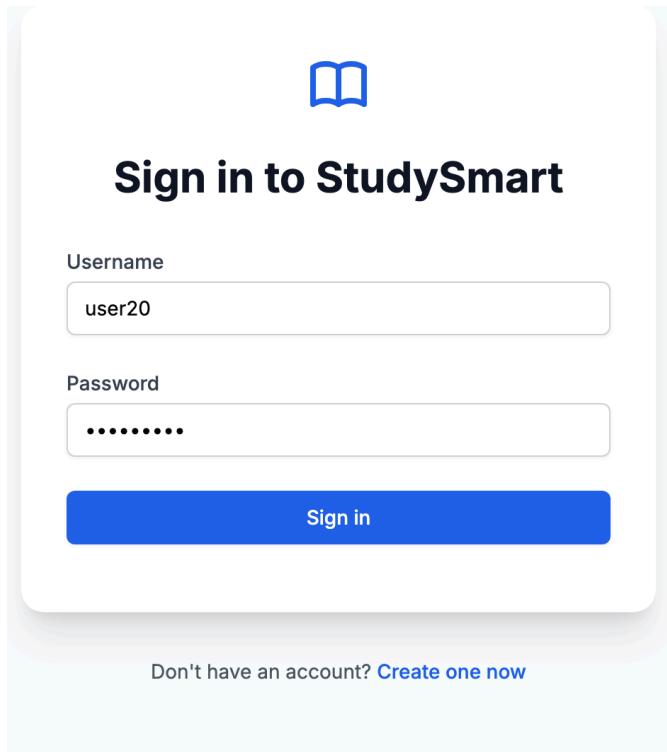
Updated user in user table in database

| | | | | | |
|--------|-----------------|-------------|-----------|------------|--|
| user20 | james@email.com | James Davis | jamespass | 2023-01-10 | |
|--------|-----------------|-------------|-----------|------------|--|

Login page:

The login form has a large blue book icon at the top. Below it, the text 'Sign in to StudySmart' is centered. There are two input fields: 'Username' and 'Password', each with a placeholder line. A large blue 'Sign in' button is positioned below the password field. At the bottom, a link 'Don't have an account? [Create one now](#)' is visible.

To login to the profile you want, put in your information for your profile for username and password and press “Sign in”



This takes you to the homepage that the register form takes you to. There are also personal statistics in the bottom right for how many courses the user is enrolled in, upcoming tasks, and the amount of study hours in the current week.

 A screenshot of the StudySmart homepage. At the top is a blue header bar with the logo 'StudySmart', navigation links 'Courses', 'Tasks', 'Study Sessions', and 'Extracurriculars', and a 'Profile' link. Below the header is a welcome message 'Welcome back, user20!'. There are six main buttons arranged in a grid: 'My Profile' (blue person icon), 'My Courses' (green document icon), 'Study Sessions' (purple clock icon), 'Extracurriculars' (orange puzzle piece icon), 'My Tasks' (red checklist icon), and 'My Stats' (grey bar chart icon). The 'My Stats' section displays the following data:

| | |
|------------------------|-------------|
| Courses Enrolled: | 0 |
| Upcoming Tasks: | 0 |
| Study Hours This Week: | 12.5 |

MyTask page:

The screenshot shows a web-based task management application. At the top, there's a header with the title "Manage My Tasks". Below the header is a blue button labeled "Add New Task". Underneath the button are two dropdown menus: "Sort by: None" and "Filter Status: All". The main area is titled "My Tasks" and contains a table with the following data:

| Task Name | Due Date | Priority | Status | Type | Actions |
|-----------------------------------|------------|----------|-----------|---------------|-------------------------|
| Final Demo Presentation CS-157A | 2024-12-05 | High | Completed | Group Task | <button>Finish</button> |
| HW8 MATH-33LA | 2024-12-09 | Low | Overdue | Personal Task | <button>Finish</button> |
| Final Exam Math33LA | 2024-12-11 | Medium | Overdue | Exam | <button>Finish</button> |
| Second Code review project CS157A | 2024-11-21 | High | Overdue | Group Task | <button>Finish</button> |
| Presentation demo | 2024-12-04 | Low | Overdue | Group Task | <button>Finish</button> |

To add a new task, simply press on the “Add New Task button”

The screenshot shows a modal dialog titled "Add New Task". It contains the following fields:

- Task Name:** demoTask
- Description:** this is the description
- Due Date:** 12/15/2024
- Priority:** Low
- Status:** Pending
- Type:** Personal Task

At the bottom right of the modal is a blue button labeled "Add Task".

Then fill out details, then press on “Add Task”

Then the list will be updated and also the database

| TASK NAME | DUE DATE | PRIORITY | STATUS | TYPE | ACTIONS |
|-----------------------------------|------------|----------|-----------|---------------|-------------------------|
| Final Demo Presentation CS-157A | 2024-12-05 | High | Completed | Group Task | <button>Finish</button> |
| HW8 MATH-33LA | 2024-12-09 | Low | Overdue | Personal Task | <button>Finish</button> |
| Final Exam Math33LA | 2024-12-11 | Medium | Overdue | Exam | <button>Finish</button> |
| Second Code review project CS157A | 2024-11-21 | High | Overdue | Group Task | <button>Finish</button> |
| Presentation demo | 2024-12-04 | Low | Overdue | Group Task | <button>Finish</button> |
| demoTask | 2024-12-15 | Low | Pending | Personal Task | <button>Finish</button> |

| Result Grid | | | | | | | |
|-------------|-----------------------------------|---|------------|----------|-----------|---------------|--------------|
| task_id | task_name | description | due_date | priority | status | type | quick_note |
| 18 | gRPC hash table program CS149 | Complete the modification in java | 2024-12-06 | Medium | Completed | Personal Task | this is note |
| 19 | HW8 MATH-33LA | finish all problems | 2024-12-09 | Low | Overdue | Personal Task | NULL |
| 20 | Final Exam Math33LA | Review all problems and practices | 2024-12-11 | Medium | Overdue | Exam | NULL |
| 21 | Second Code review project CS157A | Finish 70% of all functional requirements | 2024-11-21 | High | Overdue | Group Task | NULL |
| 22 | Presentation demo | demo for group project | 2024-12-04 | Low | Overdue | Group Task | NULL |
| 23 | Math Quiz 2 | Chapter 3-4 quiz | 2024-12-20 | Medium | Pending | Exam | NULL |
| 24 | Physics Lab | Lab report 3 | 2024-12-18 | High | Pending | Assignment | NULL |
| 25 | Study Group | Weekly meeting | 2024-12-15 | Low | Pending | Group Task | NULL |
| 26 | demoTask | this is the description | 2024-12-15 | Low | Pending | Personal Task | NULL |
| * | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

To delete the task, simply press on the “finish” button on the right side and then the task will be remove

| | | | | | |
|----------|------------|-----|---------|---------------|-------------------------|
| demoTask | 2024-12-15 | Low | Pending | Personal Task | <button>Finish</button> |
|----------|------------|-----|---------|---------------|-------------------------|

Database after delete the task:

| | | | | | | | |
|----|-------------|----------------|------------|------|---------|------------|------|
| 24 | Physics Lab | Lab report 3 | 2024-12-18 | High | Pending | Assignment | NULL |
| 25 | Study Group | Weekly meeting | 2024-12-15 | Low | Pending | Group Task | NULL |

To sort the task, we can sort it by DueDate:

The screenshot shows a "Manage My Tasks" interface. At the top, there is a "Sort by" dropdown set to "Due Date" and a "Filter Status" dropdown set to "All". Below this is a table titled "My Tasks" with the following columns: TASK NAME, DUE DATE, PRIORITY, STATUS, TYPE, and ACTIONS. The tasks listed are:

| TASK NAME | DUE DATE | PRIORITY | STATUS | TYPE | ACTIONS |
|-----------------------------------|------------|----------|---------|---------------|-------------------------|
| Second Code review project CS157A | 2024-11-21 | High | Overdue | Group Task | <button>Finish</button> |
| Presentation demo | 2024-12-04 | Low | Overdue | Group Task | <button>Finish</button> |
| Final Demo Presentation CS-157A | 2024-12-05 | High | Overdue | Group Task | <button>Finish</button> |
| HW8 MATH-33LA | 2024-12-09 | Low | Overdue | Personal Task | <button>Finish</button> |
| Final Exam Math33LA | 2024-12-11 | Medium | Overdue | Exam | <button>Finish</button> |

Or by Priority:

The screenshot shows a "Manage My Tasks" interface. At the top, there is a "Sort by" dropdown set to "Priority" and a "Filter Status" dropdown set to "All". Below this is a table titled "My Tasks" with the following columns: TASK NAME, DUE DATE, PRIORITY, STATUS, TYPE, and ACTIONS. The tasks listed are:

| TASK NAME | DUE DATE | PRIORITY | STATUS | TYPE | ACTIONS |
|-----------------------------------|------------|----------|---------|---------------|-------------------------|
| Final Demo Presentation CS-157A | 2024-12-05 | High | Overdue | Group Task | <button>Finish</button> |
| Second Code review project CS157A | 2024-11-21 | High | Overdue | Group Task | <button>Finish</button> |
| Final Exam Math33LA | 2024-12-11 | Medium | Overdue | Exam | <button>Finish</button> |
| HW8 MATH-33LA | 2024-12-09 | Low | Overdue | Personal Task | <button>Finish</button> |
| Presentation demo | 2024-12-04 | Low | Overdue | Group Task | <button>Finish</button> |

Or Filter by the status like Pending:

The screenshot shows the 'Manage My Tasks' page of the StudySmart application. At the top, there is a blue header bar with the 'StudySmart' logo, navigation links for 'Courses', 'Tasks', 'Study Sessions', and 'Extracurriculars', and user links for 'Profile' and 'Logout'. Below the header, a breadcrumb navigation shows a back arrow and the current page title 'Manage My Tasks'. A prominent blue button labeled 'Add New Task' is located above a search bar. The search bar includes dropdowns for 'Sort by: None' and 'Filter Status: Pending'. The main content area is titled 'My Tasks' and contains a table with the following data:

| TASK NAME | DU DATE | PRIORITY | STATUS | TYPE | ACTIONS |
|---------------------|------------|----------|---------|------------|-------------------------|
| Submit the report | 2024-12-16 | Low | Pending | Group Task | <button>Finish</button> |
| Final Exam CS-157A | 2024-12-16 | Medium | Pending | Exam | <button>Finish</button> |
| Final Exam Math33LA | 2024-12-18 | Medium | Pending | Exam | <button>Finish</button> |

To go to taskWorkstation page, simply click on any task:

The screenshot shows a "Task Manager" interface with the following sections:

- Task Description:** A text area containing "Merge and fix all the conflict". Below it is a blue button labeled "Update Description".
- Notes:** A text area with placeholder text "Add your notes here...". Below it is a blue button labeled "Update Notes".
- Resources:** A section with a "+ Add Resource" button.
- Related Course:** A dropdown menu labeled "Select a course" and a blue "Link Course" button.
- Progress:** A progress bar with five circular markers at 0%, 25%, 50%, 75%, and 100%.

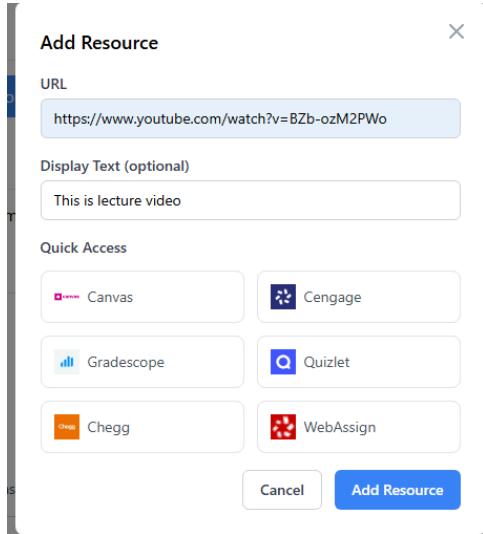
At the top right, there are buttons for "Start Timer" and "Play Music". The title of the task is "Final Demo Presentation CS-157A" and the due date is "Due: 2024-12-05".

Here is the current row in database for this task:

| | | | | | | | |
|---------------------------------|---------------------------------|------------|--------|-----------|------------|------|-----|
| Final Demo Presentation CS-157A | Merge and fix all the conflict | 2024-12-05 | High | Completed | Group Task | NULL | 0 |
| Final Demo Presentation CS-157A | Final Demo Presentation CS-157A | 2024-12-05 | Medium | Completed | Group Task | NULL | 100 |

To update description simply type in the description field and press on update description. Similarly with the note, you can update the note that you have.

To add a resource simply press on “Add Resource button”, then a modal will pop up to fill in

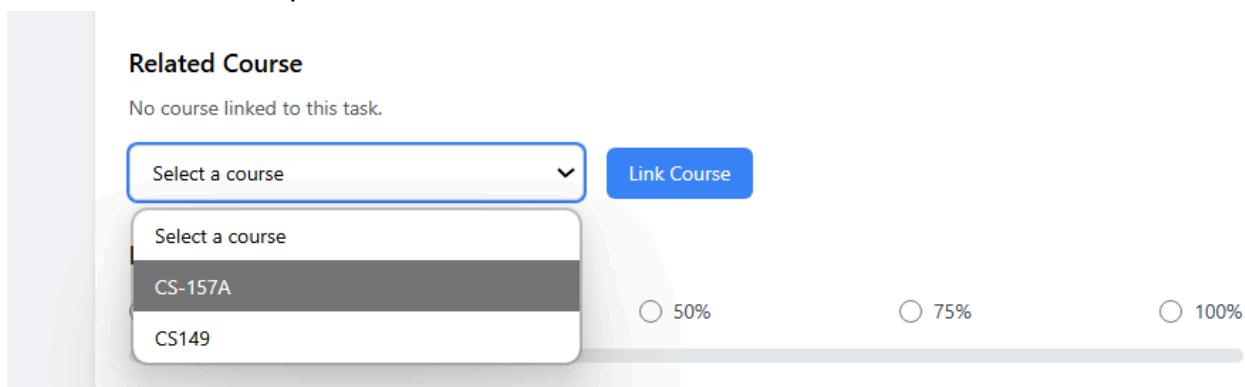


You can click on any quick access link below to automate fill in with the url of the corresponding website.

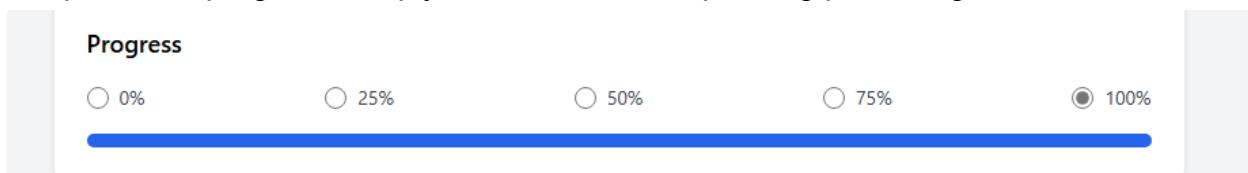
You can update the display text to know the name of the resource

After that, press “Add Resource” to update

You can also link a course that you have to know which course is related to the task by select a course and press “link course”



To update the progress, simply choose the corresponding percentage



Here is the final result after all the updates to the task:

The screenshot shows a task management interface with the following details:

- Task Description:** Updated Task
- Notes:** Updated note for demo
- Resources:** This is lecture video
- Related Course:** Currently linked to: CS-157A
 - Unlink Course button
- Progress:** 100% completed

The table:

| | | | | | | | | |
|----|---------------------------------|-----------------------------------|------------|--------|-----------|---------------|------------------------------------|-----|
| 7 | Fight with chihuahua | chihuahua is a b*tch | 2024-11-21 | High | Overdue | Assignment | why is the chihuahua so aggressive | 50 |
| 10 | testingTask | task task task tsaK KSI | 2024-11-21 | Low | Overdue | Assignment | im working harrddd????? | 50 |
| 17 | Final Demo Presentation CS-157A | Updated Task | 2024-12-05 | High | Completed | Group Task | Updated note for demo | 100 |
| 18 | gRPC hash table program CS149 | Complete the modification in java | 2024-12-06 | Medium | Completed | Personal Task | this is note | 100 |
| 19 | HW8 MATH-33LA | finish all problems | 2024-12-09 | Low | Overdue | Personal Task | HULL | 0 |
| 20 | 19 Exam Math33LA | Review all problems and practices | 2024-12-11 | Medium | Overdue | Exam | HULL | 0 |

The link task with course table:

The screenshot shows two tables:

| course_id | task_id |
|-----------|---------|
| 1 | 17 |
| 1 | 18 |
| 2 | 23 |
| 3 | 24 |
| 4 | 25 |
| * | HULL |

| course_id | name | instructor | start_date | end_date |
|-----------|-----------|---------------|------------|------------|
| 1 | CS-157A | Mike Wu | 2024-09-14 | 2024-12-18 |
| 2 | MATH-33LA | John Smith | 2024-01-08 | 2024-12-20 |
| 3 | PHYS-50 | Sarah Johnson | 2024-01-08 | 2024-12-20 |
| 4 | CS149 | Sriram Rao | 2024-01-08 | 2024-12-20 |
| 5 | CS-146 | David Lee | 2024-09-14 | 2024-12-18 |
| 6 | ENGL-1A | Emily Brown | 2024-01-08 | 2024-12-20 |
| 7 | CHEM-1A | Michael Chen | 2024-01-08 | 2024-12-20 |

We can also see that the task is mark as complete status when we update the progress to 100%

| TASK NAME | DU DATE | PRIORITY | STATUS | TYPE | ACTIONS |
|-----------------------------------|------------|----------|-----------|---------------|-------------------------|
| Final Demo Presentation CS-157A | 2024-12-05 | High | Completed | Group Task | <button>Finish</button> |
| HW8 MATH-33LA | 2024-12-09 | Low | Overdue | Personal Task | <button>Finish</button> |
| Final Exam Math33LA | 2024-12-11 | Medium | Overdue | Exam | <button>Finish</button> |
| Second Code review project CS157A | 2024-11-21 | High | Overdue | Group Task | <button>Finish</button> |
| Presentation demo | 2024-12-04 | Low | Overdue | Group Task | <button>Finish</button> |

On the navbar on top there is a “start timer” button to record study sessions and “Stop timer button” after finishing recording.

The “Play Music” button is for playing some music and we can adjust the audio with a fader next to it.

[← Back to Task List](#)
Task Manager
Start Timer
Play Music
—

Final Demo Presentation CS-157A Due: 2024-12-05

[← Back to Task List](#)
Task Manager
Stop Timer
00:00:52
Play Music
—

The table after recording the time:

A screenshot of the MySQL Workbench interface. At the top, there is a toolbar with various icons. Below the toolbar, a query editor window shows the SQL command: "1 • SELECT * FROM studysmart.studysession;". The main area displays the results of this query in a "Result Grid". The grid has four columns: "session_id", "start_time", "end_time", and "date_recorded". The data consists of eight rows, each representing a study session. The sessions are listed from oldest to newest. The newest session (session_id 49) has a start time of 18:36:18 and an end time of 18:37:13, with a duration of approximately 50 seconds.

| | session_id | start_time | end_time | date_recorded |
|--|------------|------------|----------|---------------|
| | 42 | 18:19:01 | 18:19:17 | 2024-12-03 |
| | 43 | 09:00:00 | 11:00:00 | 2024-12-03 |
| | 44 | 14:00:00 | 16:00:00 | 2024-12-03 |
| | 45 | 23:55:28 | 23:55:33 | 2024-12-05 |
| | 46 | 16:40:24 | 16:40:32 | 2024-12-06 |
| | 47 | 18:34:05 | 18:34:38 | 2024-12-16 |
| | 48 | 18:36:00 | 18:36:11 | 2024-12-16 |
| | 49 | 18:36:18 | 18:37:13 | 2024-12-16 |
| | NULL | NULL | NULL | NULL |

The newest recording is the session_id 49 with roughly 50 seconds from start to end time

MyProfile page:

User Profile



Username
user20

Name
James Davis

Email
james@email.com

Password
..... ⚡

[Edit Profile](#)

[Logout](#)

To edit the user profile click on “Edit Profile”

< Back

User Profile

Profile Picture

No file chosen

Name

James Davis

Email

james@email.com

New Password

Enter new password (leave blank to keep current)

Confirm New Password

Confirm new password

[Save Changes](#)

[Cancel](#)

Fill out any details you want to change

< Back

User Profile

Profile Picture

No file chosen

Name

James Preston

Email

james.p@email.com

New Password

Enter new password (leave blank to keep current)

Confirm New Password

Confirm new password

Then press “Save Changes”

If done correctly, a message will appear saying “Profile Updated Successfully” and the values will change

Profile Updated Successfully!

< Back

User Profile



Username
user20

Name
James Preston

Email
james.p@email.com

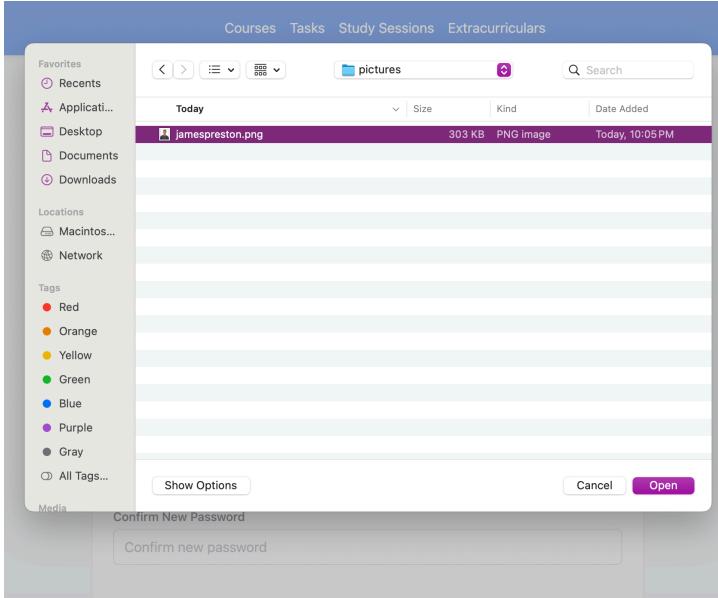
Password
..... @

Updated entry in database

user20 james.p@email.com James Preston jamespass 2023-01-10

NULL

Uploading a profile picture, go to edit profile and click choose file



After the file is selected the name of the file will be displayed, press save changes

A screenshot of a 'User Profile' edit form. It includes fields for 'Profile Picture' (with a 'Choose File' button showing 'jamespreston.png'), 'Name' (set to 'James Preston'), 'Email' (set to 'james.p@email.com'), 'New Password' (placeholder 'Enter new password (leave blank to keep current)'), 'Confirm New Password' (placeholder 'Confirm new password'), and two buttons at the bottom: 'Save Changes' (blue) and 'Cancel'.

The result will be that the profile picture will be updated and will display “Profile Updated Successfully”

Profile Updated Successfully!

[Back](#)

User Profile



Username
user20

Name
James Preston

Email
james.p@email.com

Password
..... ⚡

[Edit Profile](#)

Updated entry in database

user20 james.p@email.com James Preston jamespass 2023-01-10 uploads/jamespreston.png

My Courses Page along with Assignments and Grades

When you click the course tab you will be brought to the courses page

[StudySmart](#)

Courses Tasks Study Sessions Extracurriculars

[Profile Logout](#)

[Back](#)

Manage My Courses

[Add New Course](#)

My Courses

| COURSE NAME | INSTRUCTOR | GRADE PERCENTAGE | LETTER GRADE | START DATE | END DATE | ASSIGNMENTS | REMOVE COURSE |
|----------------------------------|----------------|------------------|--------------|------------|------------|-----------------------------|------------------------|
| Introduction to Computer Science | Dr. Smith | 99.955 | A | 2023-09-01 | 2023-12-15 | Assignments | Remove |
| CS157A | Dr. Wu | 0.0 | N/A | 2024-12-02 | 2024-12-31 | Assignments | Remove |
| CS166 | Dr. Targ | 0.0 | N/A | 2024-12-05 | 2025-01-03 | Assignments | Remove |
| CS777 | Dr. Jackpot | 0.0 | N/A | 2024-12-07 | 2025-07-07 | Assignments | Remove |
| CS 999 | Professor Nine | 0.0 | N/A | 2024-12-09 | 2025-01-09 | Assignments | Remove |

To create a new course, Click the blue “Add New Course” Button

This will bring up a new screen for you to put the information into:

Add New Course

X

Course Name:

Enter course name

Instructor:

Enter instructor name

Start Date:

mm/dd/yyyy



End Date:

mm/dd/yyyy



Add Course

Now adding some info:

Add New Course

X

Course Name:

Newest Course

Instructor:

Dr. New

Start Date:

12/05/2024



End Date:

01/09/2025



Add Course

Click Add Course:

Now the newest course will be added to the bottom of the table

| My Courses | | | | | | | |
|----------------------------------|----------------|------------------|--------------|------------|------------|-------------|---------------|
| COURSE NAME | INSTRUCTOR | GRADE PERCENTAGE | LETTER GRADE | START DATE | END DATE | ASSIGNMENTS | REMOVE COURSE |
| Introduction to Computer Science | Dr. Smith | 99.955 | A | 2023-09-01 | 2023-12-15 | Assignments | Remove |
| CS157A | Dr. Wu | 0.0 | N/A | 2024-12-02 | 2024-12-31 | Assignments | Remove |
| CS166 | Dr. Tarng | 0.0 | N/A | 2024-12-05 | 2025-01-03 | Assignments | Remove |
| CS777 | Dr. Jackpot | 0.0 | N/A | 2024-12-07 | 2025-07-07 | Assignments | Remove |
| CS 999 | Professor Nine | 0.0 | N/A | 2024-12-09 | 2025-01-09 | Assignments | Remove |
| Newest Course | Dr. New | 0.0 | N/A | 2024-12-05 | 2025-01-09 | Assignments | Remove |

Here is the data added to the “courses” table:

(Note: Intro to Computer Science is in the table, it is just cut off in the screenshot)

| course_id | name | instructor | start_date | end_date |
|-----------|---------------|----------------|------------|------------|
| 20 | CS157A | Dr. Wu | 2024-12-02 | 2024-12-31 |
| 21 | CS166 | Dr. Tarng | 2024-12-05 | 2025-01-03 |
| 22 | CS777 | Dr. Jackpot | 2024-12-07 | 2025-07-07 |
| 23 | CS 999 | Professor Nine | 2024-12-09 | 2025-01-09 |
| 24 | Newest Course | Dr. New | 2024-12-05 | 2025-01-09 |

And as we can see the “enrolled” table is updated too:

(same thing, we have 6 courses but cut off 2 of them in screenshot)

| user_id | course_id |
|---------|-----------|
| user1 | 21 |
| user1 | 22 |
| user1 | 23 |
| user1 | 24 |
| NULL | NULL |

Now if we enter into the “Newest Course” assignments:

We will be greeted with this page. It is empty so lets add some new assignments to it:

Click the blue “Add New Assignment” button:

Add New Assignments X

Assignment Name:
Assignment1

Description:
assi1

Grade:
77

Maximum Possible Grade:
100

Weight:
11

Add Assignment

And fill it with the information for the new course.

Now click Add Assignment to add it:

Manage My Assignments For Newest Course

Add New Assignment

My Assignments

| Assignment Name | Description | Grade | Max Grade | Weight (%) | Remove |
|-----------------|-------------|-------|-----------|------------|-------------------------|
| Assignment1 | assi1 | 77.0 | 100.0 | 11.0 | <button>Remove</button> |

The Newest added assignment is now shown

The “assignment” table is also updated:

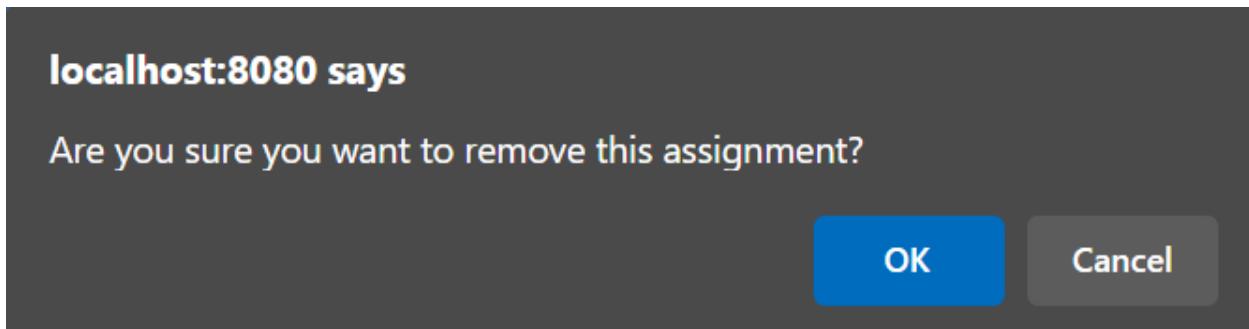
| assignment_id | name | description | grade | max_grade | weight |
|---------------|-------------|-------------|-------|-----------|--------|
| 20 | Assignment1 | assi1 | 77 | 100 | 11 |

And the “assignedBy” table has an update too:

| assignment_id | course_id |
|---------------|-----------|
| 20 | 24 |

We can see that the assignedBy is linked to assignments and courses table by ids

Now if you remove the assignment then you will be greeted with this confirmation:



Here is the assignment table **before** deletion:

| | | | | | |
|------|------------------------|-----------------------------------|------|------|------|
| 14 | Business Case Analysis | Analysis of a business case study | 86 | 100 | 0.3 |
| 15 | Music Composition | Compose a short musical piece | 51 | 100 | 0.25 |
| 20 | Assignment1 | assi1 | 77 | 100 | 11 |
| NULL | NULL | NULL | NULL | NULL | NULL |

Here is the assignedBy table **before** deletion:

| | |
|------|------|
| 14 | 14 |
| 15 | 15 |
| 20 | 24 |
| NULL | NULL |

Assignment table **after** deletion:

| | | | | | |
|------|------------------------|-----------------------------------|------|------|------|
| 14 | Business Case Analysis | Analysis of a business case study | 86 | 100 | 0.3 |
| 15 | Music Composition | Compose a short musical piece | 51 | 100 | 0.25 |
| NULL | NULL | NULL | NULL | NULL | NULL |

AssignedBy table **after** deletion:

| | |
|------|------|
| 14 | 14 |
| 15 | 15 |
| NULL | NULL |

Now lets go back and delete the course we originally made:

localhost:8080 says

Are you sure you want to remove this course?

OK Cancel

Manage My Courses

Add New Course

My Courses

| COURSE NAME | INSTRUCTOR | GRADE PERCENTAGE | LETTER GRADE | START DATE | END DATE | ASSIGNMENTS | REMOVE COURSE |
|----------------------------------|----------------|------------------|--------------|------------|------------|------------------------------|-------------------------|
| Introduction to Computer Science | Dr. Smith | 99.955 | A | 2023-09-01 | 2023-12-15 | <button>Assignments</button> | <button>Remove</button> |
| CS157A | Dr. Wu | 0.0 | N/A | 2024-12-02 | 2024-12-31 | <button>Assignments</button> | <button>Remove</button> |
| CS166 | Dr. Tarn | 0.0 | N/A | 2024-12-05 | 2025-01-03 | <button>Assignments</button> | <button>Remove</button> |
| CS777 | Dr. Jackpot | 0.0 | N/A | 2024-12-07 | 2025-07-07 | <button>Assignments</button> | <button>Remove</button> |
| CS 999 | Professor Nine | 0.0 | N/A | 2024-12-09 | 2025-01-09 | <button>Assignments</button> | <button>Remove</button> |
| Newest Course | Dr. New | 0.0 | N/A | 2024-12-05 | 2025-01-09 | <button>Assignments</button> | <button>Remove</button> |

Result:

| Manage My Courses | | | | | | | |
|----------------------------------|----------------|------------------|--------------|------------|------------|-------------|---------------|
| My Courses | | | | | | | |
| COURSE NAME | INSTRUCTOR | GRADE PERCENTAGE | LETTER GRADE | START DATE | END DATE | ASSIGNMENTS | REMOVE COURSE |
| Introduction to Computer Science | Dr. Smith | 99.955 | A | 2023-09-01 | 2023-12-15 | Assignments | Remove |
| CS157A | Dr. Wu | 0.0 | N/A | 2024-12-02 | 2024-12-31 | Assignments | Remove |
| CS166 | Dr. Tarng | 0.0 | N/A | 2024-12-05 | 2025-01-03 | Assignments | Remove |
| CS777 | Dr. Jackpot | 0.0 | N/A | 2024-12-07 | 2025-07-07 | Assignments | Remove |
| CS 999 | Professor Nine | 0.0 | N/A | 2024-12-09 | 2025-01-09 | Assignments | Remove |

Courses table before:

Here is the data added to the “courses” table:

| course_id | name | instructor | start_date | end_date |
|-----------|---------------|----------------|------------|------------|
| 20 | CS157A | Dr. Wu | 2024-12-02 | 2024-12-31 |
| 21 | CS166 | Dr. Tarng | 2024-12-05 | 2025-01-03 |
| 22 | CS777 | Dr. Jackpot | 2024-12-07 | 2025-07-07 |
| 23 | CS 999 | Professor Nine | 2024-12-09 | 2025-01-09 |
| 24 | Newest Course | Dr. New | 2024-12-05 | 2025-01-09 |

And as we can see the “enrolled” table is updated too:

| user_id | course_id |
|---------|-----------|
| user1 | 21 |
| user1 | 22 |
| user1 | 23 |
| user1 | 24 |
| NULL | NULL |

Courses table after:

| | | | | |
|------|--------|----------------|------------|------------|
| 20 | CS157A | Dr. Wu | 2024-12-02 | 2024-12-31 |
| 21 | CS166 | Dr. Tarng | 2024-12-05 | 2025-01-03 |
| 22 | CS777 | Dr. Jackpot | 2024-12-07 | 2025-07-07 |
| 23 | CS 999 | Professor Nine | 2024-12-09 | 2025-01-09 |
| NULL | NULL | NULL | NULL | NULL |

Enrolled table after:

| | | |
|---|--------------|------|
| | user1 | 21 |
| | user1 | 22 |
| | user1 | 23 |
| * | NULL | NULL |

We can see that courseID 24 is removed from all tables

Now lets add a new course with assignments and see how the grade fluctuates:

Manage My Courses

Add New Course

My Courses

| COURSE NAME | INSTRUCTOR | GRADE PERCENTAGE | LETTER GRADE | START DATE | END DATE | ASSIGNMENTS | REMOVE COURSE |
|----------------------------------|------------------|------------------|--------------|------------|------------|--------------------|---------------|
| Introduction to Computer Science | Dr. Smith | 99.955 | A | 2023-09-01 | 2023-12-15 | Assignments | Remove |
| CS157A | Dr. Wu | 0.0 | N/A | 2024-12-02 | 2024-12-31 | Assignments | Remove |
| CS166 | Dr. Targ | 0.0 | N/A | 2024-12-05 | 2025-01-03 | Assignments | Remove |
| CS777 | Dr. Jackpot | 0.0 | N/A | 2024-12-07 | 2025-07-07 | Assignments | Remove |
| CS 999 | Professor Nine | 0.0 | N/A | 2024-12-09 | 2025-01-09 | Assignments | Remove |
| Grade Testing | Dr. Grade Tester | 0.0 | N/A | 2024-12-02 | 2024-12-31 | Assignments | Remove |

Courses without assignments will have a grade of N/A until you add an assignment

Add a new assignment:

Manage My Assignments For Grade Testing

Add New Assignment

My Assignments

| ASSIGNMENT NAME | DESCRIPTION | GRADE | MAX GRADE | WEIGHT (%) | REMOVE |
|-----------------|-------------|-------|-----------|------------|---------------|
| Assignment1 | assi1 | 77.0 | 100.0 | 11.0 | Remove |

Now we can see the course has a grade of 97.47% = A

Manage My Courses

Add New Course

My Courses

| COURSE NAME | INSTRUCTOR | GRADE PERCENTAGE | LETTER GRADE | START DATE | END DATE | ASSIGNMENTS | REMOVE COURSE |
|----------------------------------|------------------|------------------|--------------|------------|------------|-------------|---------------|
| Introduction to Computer Science | Dr. Smith | 99.955 | A | 2023-09-01 | 2023-12-15 | Assignments | Remove |
| CS157A | Dr. Wu | 0.0 | N/A | 2024-12-02 | 2024-12-31 | Assignments | Remove |
| CS166 | Dr. Targ | 0.0 | N/A | 2024-12-05 | 2025-01-03 | Assignments | Remove |
| CS777 | Dr. Jackpot | 0.0 | N/A | 2024-12-07 | 2025-07-07 | Assignments | Remove |
| CS 999 | Professor Nine | 0.0 | N/A | 2024-12-09 | 2025-01-09 | Assignments | Remove |
| Grade Testing | Dr. Grade Tester | 97.47 | A | 2024-12-02 | 2024-12-31 | Assignments | Remove |

The Grade table:

| | grade_id | decimal_grade | letter_grade | date_recorded |
|---|----------|---------------|--------------|---------------|
| ▶ | 1 | 99.955 | A | 2023-10-15 |
| | 2 | 99.2 | A | 2023-10-16 |
| | 3 | 97.6 | A | 2023-10-17 |
| | 4 | 96.7 | A | 2023-10-18 |
| | 5 | 98.75 | A | 2023-10-19 |
| | 6 | 98.2 | A | 2023-10-20 |
| | 7 | 98 | A | 2023-10-21 |
| | 8 | 96.1 | A | 2023-10-22 |
| | 9 | 98.6 | A | 2023-10-23 |
| | 10 | 97.75 | A | 2023-10-24 |
| | 11 | 96 | A | 2023-10-25 |
| | 12 | 98 | A | 2023-10-26 |
| | 13 | 97.6 | A | 2023-10-27 |
| | 14 | 95.8 | A | 2023-10-28 |
| | 15 | 87.75 | B | 2023-10-29 |
| | 20 | 0 | N/A | 2024-12-02 |
| | 21 | 0 | N/A | 2024-12-05 |
| | 22 | 0 | N/A | 2024-12-07 |
| | 23 | 0 | N/A | 2024-12-09 |
| * | 25 | 97.47 | A | 2024-12-02 |
| * | NULL | NULL | NULL | NULL |

Note: The course we deleted earlier had ID 24 so this course has ID 25

“gradeOf” table:

| | grade_id | course_id |
|---|----------|-----------|
| ▶ | 1 | 1 |
| | 2 | 2 |
| | 3 | 3 |
| | 4 | 4 |
| | 5 | 5 |
| | 6 | 6 |
| | 7 | 7 |
| | 8 | 8 |
| | 9 | 9 |
| | 10 | 10 |
| | 11 | 11 |
| | 12 | 12 |
| | 13 | 13 |
| | 14 | 14 |
| | 15 | 15 |
| | 20 | 20 |
| | 21 | 21 |
| | 22 | 22 |
| | 23 | 23 |
| * | 25 | 25 |
| | NULL | NULL |



We can see that course_id 25 pairs with our grade_id

Adding a new assignment (assignment2) will update the grade table:

| My Assignments | | | | | |
|-----------------|-------------|-------|-----------|------------|-------------------------|
| ASSIGNMENT NAME | DESCRIPTION | GRADE | MAX GRADE | WEIGHT (%) | REMOVE |
| Assignment1 | assi1 | 77.0 | 100.0 | 11.0 | <button>Remove</button> |
| Assignment2 | assi1 | 99.0 | 100.0 | 11.0 | <button>Remove</button> |

| | | | |
|----|-------|-----|------------|
| 23 | 0 | N/A | 2024-12-09 |
| 25 | 97.36 | A | 2024-12-02 |

We can see the new assignment changed the grade from 97.47% to 97.36%

Study Session Stats Page



Users can also see how long they have studied for each day with stats based on the week. The user can also change increment or decrement which week they want to look at with the date being displayed in the top middle of the screen.

(Note: the current data for study sessions is not up to the current week, please navigate to the week like in the image above for all UI features)

| Result Grid | | | | |
|----------------|------------|------------|----------|---------------|
| | session_id | start_time | end_time | date_recorded |
| 7 | 7 | 19:00:00 | 21:00:00 | 2024-12-01 |
| 8 | 8 | 10:00:00 | 12:30:00 | 2024-12-01 |
| 9 | 9 | 15:00:00 | 17:30:00 | 2024-11-30 |
| 10 | 10 | 09:30:00 | 12:00:00 | 2024-11-29 |
| 11 | 11 | 14:00:00 | 16:30:00 | 2024-11-29 |
| 12 | 12 | 10:00:00 | 11:30:00 | 2024-11-28 |
| 13 | 13 | 13:00:00 | 14:30:00 | 2024-11-27 |
| 14 | 14 | 16:00:00 | 18:00:00 | 2024-11-27 |
| 15 | 15 | 09:00:00 | 11:00:00 | 2024-11-25 |
| 16 | 16 | 14:00:00 | 16:00:00 | 2024-11-25 |
| 17 | 17 | 10:00:00 | 12:00:00 | 2024-11-24 |
| 18 | 18 | 15:00:00 | 17:00:00 | 2024-11-24 |
| studvsession 1 | | | | |

MyExtracurricular Page:

< Back

Manage My Extracurriculars

Add New Extracurricular

My Extracurriculars

No extracurriculars found. Add a new extracurricular to get started!

At the start there will be no extracurriculars added, to add an extracurricular press “Add New Extracurricular”

< Back

Manage My Extracurriculars

Add New Extracurricular

My Extracurriculars

No extracurriculars found. Add a new extracurricular to get started!

Add New Extracurricular

Activity Name:

Description:

Start Date:

End Date:

Add Extracurricular

A form will show up for the extracurriculars, input all your values and press “Add Extracurricular”

Add New Extracurricular

Activity Name:
SCE Club meetings

Description:
Meeting for all SCE members

Start Date:
12/21/2024

End Date:
12/20/2025

Add Extracurricular

After the button is pressed, the extracurricular will display on the page

| ACTIVITY NAME | DESCRIPTION | START DATE | END DATE | REMOVE EXTRACURRICULAR |
|-------------------|-----------------------------|------------|------------|------------------------|
| SCE Club meetings | Meeting for all SCE members | 2024-12-21 | 2025-12-20 | Remove |

Here you can view all your extracurriculars with the activity name, description, start date, end date, and an option to remove it

Database after extracurricular is added:

7

SCE Club meetings Meeting for all SCE members 2024-12-21 2025-12-20

To remove, press the red Remove button under Remove Extracurricular

The screenshot shows a web browser window for 'StudySmart'. At the top, there's a blue header bar with the 'StudySmart' logo and navigation links for 'Profile' and 'Logout'. Below the header, the main content area has a title 'Manage My Extracurriculars' and a button 'Add New Extracurricular'. A table titled 'My Extracurriculars' lists one item: 'SCE Club meetings' with a description 'Meeting for all SCE members', starting on '2024-12-21' and ending on '2025-12-20'. To the right of each row is a red 'Remove' button. A modal dialog box is centered over the page, displaying the message 'localhost:8080 says' and 'Are you sure you want to remove this extracurricular?'. It has two buttons: 'Cancel' (light blue) and 'OK' (dark green).

A pop up will ask you to confirm if you want to delete the extracurricular or not, if you press OK it will be deleted.

Updated Extracurricular table after delete:

| extracurricular_id | activity_name | description | start_date | end_date |
|--------------------|---------------|-------------|------------|----------|
| NULL | NULL | NULL | NULL | NULL |
| | | | | |

Lesson Learned

Minh Phong Do:

I worked on the functional requirements and features of the project, including implementing CRUD operations for task management, sorting, filtering, and building a recording study sessions feature. The biggest challenge was learning how to develop this 3-tier application using Java, Servlets, Tomcat server, and MySQL; in particular, learning to use Servlets for routing pages was quite tricky, since most modern tech stacks have that built in. Eclipse provided more challenges in its complex structure and slow performance, often unable to identify the correct libraries, requiring manual inclusion of files like servlet_api.jar to ensure consistent functionality. Debugging was an integral part of development—I debugged errors using print statements in Java and in JSP, while frequently researching solutions online. I also contributed to improving the maintainability of the project by helping team members differentiate between client-side and server-side code. This experience has vastly improved my technical capabilities in debugging, problem-solving, and engaging with this particular technology stack. It has cultivated resilience in me and the resolve to overcome challenges. For future improvements, integration of the Google Docs API into the task workstation page could make the application more efficient.

Nathan Wong:

I worked on the course, assignment, and grade features of the project. This involved CRUD operations for managing each course, assignment and grade for each user. My biggest challenges were learning how to use servlets and debugging the sql queries and updates. I had to first learn how servlets worked in order to fill the jsp pages with the data that I wanted them to have. This was challenging to me because it was the first time I worked with servlets and I was not familiar with how they functioned. Another challenge for me was debugging the sql queries and updates. This was challenging because sometimes the query or update did not go through even though everything seems correct. I learned the importance of spacing and also the order of operating your updates. If you leave out a space in your sql query statement in the java string, then the query will not work. Also, if you have to update or delete something in two different tables, it is important to consider the order of which table to operate on first. This project has helped me improve my ability to create 3-tier architectures and my ability to write sql statements to operate on database tables. It also sharpened my attention to detail and how I have to have a sharp eye to spot errors and mistakes that will stop my code from working properly.

Kyle Wang:

I worked on the user profile, updating the profile, adding profile pictures and saving them, and the extracurriculars. To build these, I had to learn how to use servlets, learn sql commands, and use CRUD operations for updating users profiles, profile pictures, and extracurriculars. I had trouble getting my update profile to work because we had some bugs with inserting users into the sql database. I also had to figure out how to only update the fields that the user wanted to update instead of all the fields. The hardest part for me to learn was saving the user profile picture. The way I resolved this issue was to make a directory if one didn't already exist to hold all the pictures and have a field in the user schema that held the path to the directory and picture. I also learned how to display messages whether it be error messages or success messages to help the user understand what happened when they submitted their information. This project helped me understand the 3-tier architecture principle as well as how to develop my own webpage using sql, servlet, apache tomcat, and jsp files. It helped me with learning to do research into topics related to databases and how to connect my webpage to a database getting user input to change the webpage.

Joshua Demo:

I worked on the design and completed the ERD diagram for the project, determining necessary functional requirements for the project, designing the UI theming for the whole project, and building the functional requirements for the study sessions and the task statistics of the user. I never worked with jsp before and tomcat server so all of the java stuff was new to me. I've previously worked with html before, but beautifying the pages is always a hard task to do. I had to google and find a few libraries to make my life a bit easier, like chartJS for the graphs and TailwindCSS for easy CSS styling. The hardest part of the project was creating a consistent database for all of the teammates to use. I decided to create a mysql database utilizing docker so each of the team members' databases were consistent. I didn't use docker too much before so I had to learn how to use it.