

Databases

ACADEMIC TO-DO LIST

Francisco Cardita 97640

Pedro Ferreira 98620

P1G4



CONTENT

01

Objective

02

Requirements

03

ERD

04

Features

05

Infrastructure

06

Demo

07

To be Implemented



OBJECTIVE



Task management platform
for university classes

REQUIREMENTS

The platform allows the user to:



Create, edit and
update tasks



Add various
attributes to each
task



Define the task
visibility
(public/private)



Search for other
users

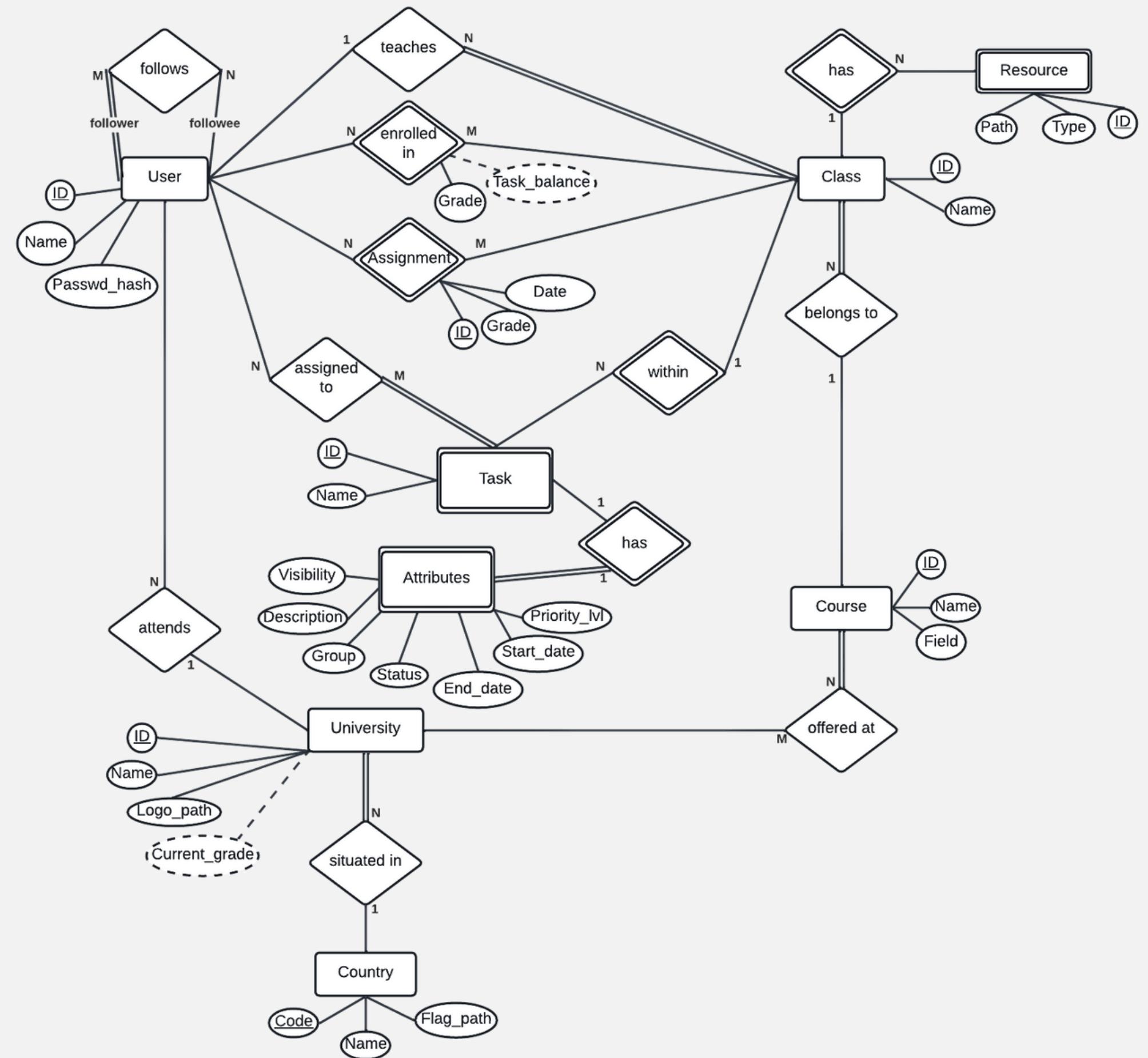


Follow users



See their
followers and
who they follow

ERD



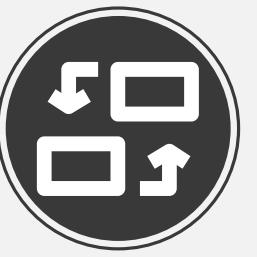
FEATURES



Triggers



Hashed
passwords



Stored
Procedures



Input validation

TRIGGERS

Soft Delete

```
CREATE TRIGGER uni_tasks.SoftDeleteTask
ON uni_tasks.task
INSTEAD OF DELETE
AS
BEGIN
    UPDATE t
    SET t.is_deleted = 1
    FROM uni_tasks.task AS t
    INNER JOIN deleted AS d ON t.id = d.id;
END
GO
```

HASHED PASSWORDS

```
-- Hash the password
DECLARE @passwordHash VARBINARY(256);
SET @passwordHash = HASHBYTES('SHA2_256', @password);

-- Store the hashed password in the database
INSERT INTO uni_tasks._user ([name], password_hash, uni_id)
VALUES (@username, @passwordHash, @uni_id);
```

STORED PROCEDURES

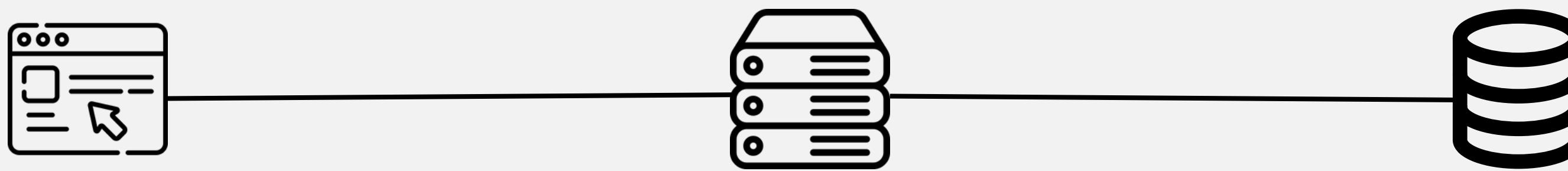
NewTask	SearchUser
UpdateTask	GetUser
DeleteTask	FollowUser
ListTasks	UnfollowUser
AssociateTaskWithUser	ListFollowers
LoginUser	ListFollowees
RegisterUser	ListUniversities
	ListClasses

INPUT VALIDATION

SQL Injection Prevention

```
EXEC uni_tasks.GetUser @usr_id=%s, [user_id]
```

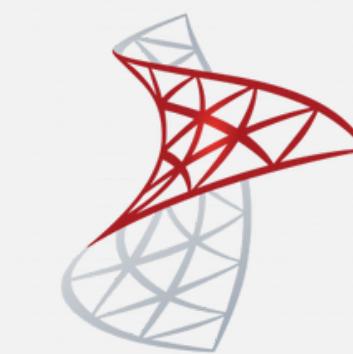
INFRASTRUCTURE



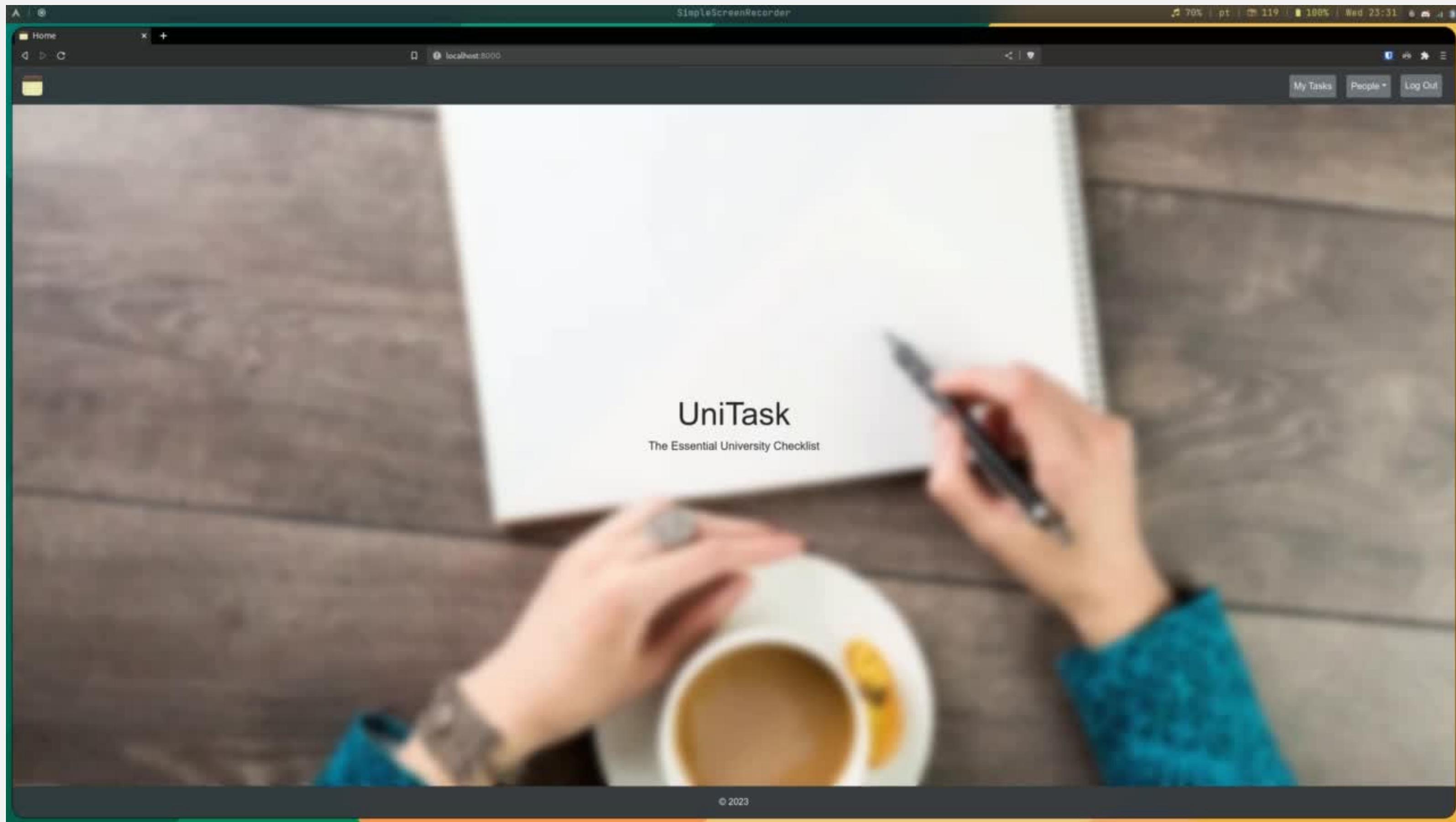
Webpage

Server

DB



DEMO



TO BE IMPLEMENTED

The user will be able to:



Login/Register



Associate other
users with tasks

THANK YOU