Markez app features:
- Membership
    - Users (students)
        - View all available courses
        - View institutions
        - View teachers profile
        - Can print course certificate
    - Teachers
        - View all available courses
        - View all students profiles
        - View only the institutions he belongs to
        - Can't create course, but can modify it and add resources.
        - Can create quiz for the course
    - Institutions
        - view only the courses that belongs to it
        - view only teachers they belongs to
        - view only students they belongs to
        - can create and modify courses
        - can send messages (DM) to teachers
    - administrator
- Courses
    - Has contents
        - Video
            - Videos must be hosted on any cloud services (ie: google drive, one drive… etc) and must be accessible to students
            - No one is permitted to download the videos.
        - Live stream via zoom
            - must be recorded
        - text
    - Has quiz
    - Has certificates
        - The students can only print the certificate when attend all course lectures.
    - Can be freemium or premium
    - Can be limited (for number of students) or unlimited
- Direct messages
    - Messages are available only between
        - Teachers & institutions
        - Students and teachers in course only
- Payment
    - Students can pay for premium courses (paid courses)
    - Institutions must pay for fees to be able to use our system
-

How markes app works

We are targeting educational **institutions** to allow them to create a platform to promote their services (mush like shopify for stores).

When an institution creates an account on our platform, they got their own page (markez.edu/<whatever>), they can add a logo, change the main color and add courses to their page (the theme is the same for all institution, only the color is changeable).

In **institution control panel,** mods can add courses, and assign it to a teacher, in order to assign a course to a teacher, the institution must add a teacher to staff first by sending a request to the teacher, then the teacher accept the request.

After adding the course data (title, cover image, price, type, category, date for live stream) to the course page, now it's the teacher's turn to add the contents for this course.

The **teachers** can receive requests from institutions (new member request, new course request if teacher is a member in this institution), after receiving and accepting a new course request, he can modify all the course's info except price, students limit and time of the course.

The teacher also can receive DMs from students only from the courses he teaches.

The **students** can view all teachers' profiles, all institutions pages, all published courses on the platform.

They can enroll to any available course, comment and rate courses they enrolled to, print certificates if they attended all lectures in a course.

Admins get all permissions to add, modify and delete all entries.

Note: this project is different from Udemy and Udacity business model, our main customers are the educational institutions.

institution

create account

↓

setup institution page

↓

lookup for teachers

↓

send request to teacher

↓

receive request from teacher

↓

create course

↓

upload course certificate (with blank names)

teachers

create account

↓

setup profile

↓

create account

↓

receive requests

↓

modify course

↓

add course content

↓

start course

students

create account

↓

setup profile

↓

lookup for courses

↓

lookup for teachers

↓

lookup for inistitutions

↓

enroll to course

↓

pay

↓

start learning

↓

finish all courses

↓

print certificate

courses

course page

response with accept or not

when course accepted

send course request

payment goes to administration to process, then it got passed to institution