

درس: مبتنی بر وب

مدرس: استاد نجاری

دانشجویان: پارسا خسروی و علی تیموری دهنو

پروژه: سیستم حضور و غیاب دانشجویان

[لینک](#) github

این پروژه یک سیستم مدیریت حضور و غیاب تحت وب است که برای دو نوع کاربر طراحی شده: **مدرس** و **دانشجو**. هدف آن، مدیریت کلاس‌ها، ثبت حضور دانشجویان، و نمایش گزارش‌های مرتبط است.

در بخش **ثبت‌نام و ورود (Authentication)**، کاربران می‌توانند با وارد کردن اطلاعاتی مانند نام، ایمیل، رمز عبور و نقش (مدرس یا دانشجو) حساب کاربری ایجاد کنند. پس از ثبت‌نام، ورود (Login) از طریق ایمیل و رمز عبور انجام می‌شود و هر کاربر فقط به بخش‌های مخصوص نقش خود دسترسی دارد. برای مثال، صفحات مخصوص مدرس فقط برای مدرسین فعال است. این کنترل دسترسی می‌تواند با **Session** پیاده‌سازی شود.

در بخش **مدیریت نقش‌ها (Role Management)**، هر نقش وظایف خاص خود را دارد:

- **دانشجو**: می‌تواند حضور خود را در کلاس‌های ثبت‌شده علامت بزند و لیست حضورهای خود را مشاهده کند.
- **مدرس**: می‌تواند کلاس جدید ایجاد کند، دانشجویان را به آن اضافه کند، حضور آن‌ها را مشاهده و ویرایش کند، و در نهایت از داده‌ها گزارش بگیرد.

در قسمت **ثبت حضور (Attendance)**، سیستم به دانشجو امکان می‌دهد در زمان برگزاری کلاس، با کلیک روی دکمه‌ای مثل "حاضر"، حضور خود را ثبت کند. همچنین تاریخ و ساعت ثبت حضور به صورت خودکار ذخیره می‌شود. مدرس نیز می‌تواند وضعیت حضور دانشجویان را ویرایش کند، مثلاً در صورت غیبت یا تأخیر.

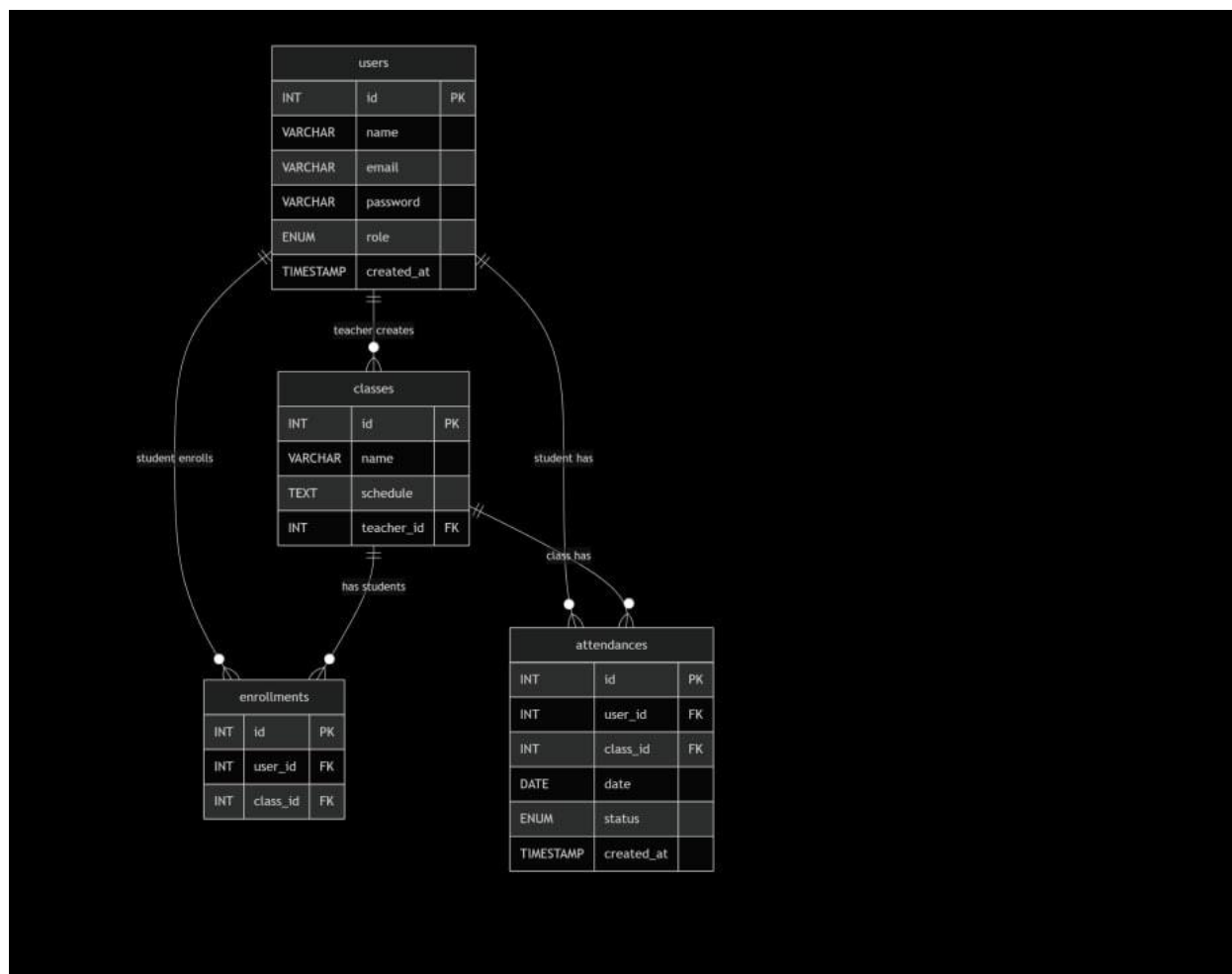
بخش **مدیریت کلاس‌ها (Class Management)** شامل ایجاد، نمایش و ویرایش کلاس‌هاست. مدرس می‌تواند جزئیاتی مانند نام کلاس، تاریخ و ساعت برگزاری و لیست دانشجویان را مشخص کند. دانشجو نیز تنها کلاس‌هایی را می‌بیند که در آن‌ها ثبت‌نام کرده است.

پس از ورود، هر کاربر وارد داشبورد خود می‌شود که اطلاعات مربوط به نقش او نمایش داده می‌شود؛ مثلاً لیست کلاس‌ها و وضعیت حضور برای دانشجو، و گزارش‌های کلی کلاس‌ها برای مدرس.

تمام داده‌ها از جمله کاربران، کلاس‌ها و حضورها در پایگاه داده‌ای مانند **MySQL** ذخیره می‌شوند.

در نهایت، برای بخش ظاهری از **HTML**، **CSS** یا فریم‌ورک‌هایی مثل **Bootstrap** استفاده می‌شود تا محیطی ساده، کاربرپسند و واکنش‌گرا (**responsive**) برای استفاده در موبایل و دسکتاپ فراهم شود.

کاربرد	فناوری / ابزار	بخش پروژه
ایجاد ساختار و محتوای صفحات وب (فرم‌ها، جدول‌ها، دکمه‌ها و غیره).	HTML5	سمت کاربر (Front-End)
طراحی ظاهر صفحات، تنظیم رنگ‌ها، فونت‌ها و واکنش‌گرایی. (Responsive Design)	CSS / Bootstrap	
پردازش داده‌ها، مدیریت ورود و خروج، کنترل نقش‌ها و ارتباط با پایگاه داده.	PHP یا Python	سمت سرور (Back-End)
حفظ وضعیت ورود کاربر و کنترل دسترسی به صفحات مخصوص هر نقش (مدرس یا دانشجو).	Session	
ذخیره‌سازی اطلاعات کاربران، کلاس‌ها و حضور دانشجویان.	MySQL	پایگاه داده (Database)
اجرای محیط PHP و MySQL به صورت محلی برای تست پروژه.	XAMPP / WAMP	ابزار توسعه (Development Tools)
ویرایش و توسعه کد پروژه.	VS Code / PyCharm	
کنترل نسخه، نگهداری و اشتراک‌گذاری پروژه.	Git & GitHub	



۱. جدول users کاربران)

توضیح: این جدول برای ذخیره اطلاعات همه کاربران سیستم (هم دانشجو، هم مدرس) استفاده می‌شود.

id: یک شماره یکتا و خودکار برای هر کاربر

name: نام کامل کاربر

email: آدرس ایمیل که برای ورود به سیستم استفاده می‌شود

password: هش می‌شود در PHP در password_hash رمز عبور که با تابع:

role: "تعیین می‌کند کاربر "مدرس" است یا "دانشجو"

created_at: تاریخ و زمان ایجاد حساب کاربری

۲. جدول classes (کلاس‌ها)

توضیح: این جدول کلاس‌های ایجاد شده توسط مدرسین را ذخیره می‌کند.

شماره یکتا برای هر کلاس: id

نام کلاس (مثلاً "پایگاه داده پیشرفته"): name

زمان برگزاری کلاس (مثلاً "سه‌شنبه‌ها ساعت ۱۰-۱۲"): schedule

(users ارتباط با جدول) شماره مدرسی که این کلاس را ایجاد کرده: teacher_id

۳. جدول enrollments (ثبت‌نام‌ها)

توضیح: این جدول ارتباط بین دانشجویان و کلاس‌ها را مدیریت می‌کند (کدام دانشجو در کدام کلاس ثبت‌نام کرده)

شماره یکتا برای هر رکورد ثبت‌نام: id

(users از جدول) شماره دانشجو: user_id

(classes از جدول) شماره کلاس: class_id

۴. جدول attendances (حضور و غیاب)

توضیح: این جدول اصلی‌ترین بخش سیستم است و سابقه حضور دانشجویان را ذخیره می‌کند.

شماره یکتا برای هر رکورد حضور: id

شماره دانشجو: user_id

شماره کلاس: class_id

تاریخ جلسه کلاس (برای تفکیک جلسات مختلف): date

وضعیت حضور ("حاضر"، "غایب"، "تأخیر"): status

زمان دقیق ثبت حضور: created_at

روابط بین جداول: (Expanded)

1. مدرس ← کلاس‌ها

- هر مدرس می‌تواند چندین کلاس ایجاد کند.
- هر کلاس دقیقاً یک مدرس دارد.

2. دانشجو ← کلاس‌ها (از طریق ثبت‌نام)

- دانشجو‌ها با جدول **enrollments** در کلاس‌ها ثبت‌نام می‌شوند.
- هر دانشجو می‌تواند در چند کلاس شرکت کند و هر کلاس می‌تواند چند دانشجو داشته باشد.

3. دانشجو ← حضور در کلاس‌ها

- جدول **attendances** وضعیت حضور یا غیاب دانشجو در هر کلاس را ثبت می‌کند.
- می‌توان بر اساس تاریخ یا کلاس، حضورها را بررسی کرد.

مثال‌های عملی:

- ثبت حضور: وقتی دانشجو روی «حاضر» کلیک می‌کند، وضعیت حضور او در کلاس ثبت می‌شود.
- مشاهده حضور: مدرس می‌تواند لیست حضور دانشجو‌ها را برای یک کلاس مشخص و در یک تاریخ خاص ببیند.
- کلاس‌های یک دانشجو: می‌توان تمام کلاس‌هایی که یک دانشجو در آن‌ها ثبت‌نام کرده را نمایش داد.