

پروژه‌هایی

کتاب ۱۱۷



پروژه مدیریت برگزاری آنلاین آزمون

هدف از این پروژه ، پیاده سازی یک **Web Application** جهت " ایجاد و برگزاری آنلاین آزمون " می باشد.



این پروژه به طور کلی شامل **امکانات/ویژگی** های زیر می باشد:

- ✓ ثبت نام به عنوان استاد/دانشجو
- ✓ تایید ثبت نام کاربر توسط مدیر
- ✓ جستجو و فیلتر رکوردهای دانشجویان/اساتید
- ✓ تعریف دوره (Course) توسط مدیر
- ✓ افزودن استاد به دوره توسط مدیر
- ✓ افزودن دانشجو به دوره توسط مدیر
- ✓ تعریف آزمون توسط استاد
- ✓ افزودن سوالات چند گزینه ای و تشریحی به آزمون
- ✓ بانک سوالات
- ✓ تعیین نمره پیشفرض برای هر سوال در آزمون
- ✓ امکان برگزاری آزمون
- ✓ امکان نمره دهی به دانشجویان (اتوماتیک و دستی)
- ✓ نقش های مدیر، استاد و دانشجو

ثبت نام به عنوان استاد/دانشجو

باید امکانی برای دو نقش **استاد** و **دانشجو** فراهم شود که هر دو بتوانند در سیستم ثبت نام کنند. کاربر؛ در هنگام ثبت نام باید تعیین کند که در نقش دانشجو ثبت نام خواهد کرد یا استاد. بعد از ثبت نام، این کاربر، در وضعیت **"در انتظار تایید"** قرار می گیرد.

تایید ثبت نام کاربر توسط مدیر

مدیر باید لیستی از کلیه افراد ثبت نام شده، در کل سامانه به همراه اطلاعات مورد نیاز آن ها را داشته باشد. بعد از ثبت نام موفق یک کاربر، کاربر با نقش مدیر، کاربر ثبت نام شده را تایید خواهد کرد و کاربر مورد نظر به وضعیت **"تایید شده"**، تغییر می کند. مدیر همچنین، می تواند اطلاعات کاربران را ویرایش کند و نقش ایشان را تغییر دهد.

جستجو و فیلتر رکوردهای دانشجویان/اساتید

مدیر باید بتواند در لیست مورد نظر، بر اساس پارامترهای مختلفی، مثل نقش کاربر (استاد/دانشجو)، نام، نام خانوادگی و ... فیلترهای را اعمال کند و به جستجو بپردازد و کلیه ی رکوردهایی که شرایط تعیین شده را دارند به عنوان نتیجه مورد نظر، به او نمایش داده شود.

تعریف دوره (Course) توسط مدیر

کاربر با نقش مدیر، می تواند لیست کلیه دوره ها (Course) را مشاهده کند، همینطور می تواند دوره جدید تعریف کند. اطلاعات هر دوره، حد/قل شامل موارد زیر می باشد:

- عنوان دوره
- شناسه ی یکتای دوره
- تاریخ شروع دوره
- تاریخ پایان دوره
- و ...

افزودن استاد/دانشجو به دوره توسط مدیر

هر دوره، فقط یک استاد خواهد داشت که پس از ثبت نام وی در سیستم، باید توسط مدیر به عنوان استاد یک دوره خاص، به آن دوره، اضافه شود. یک استاد می توان در چند دوره تدریس کند.

از سایر اختیارات کاربری با نقش مدیر، می توان به توانایی اضافه کردن دانشجوها به یک دوره خاص اشاره کرد.

قابلیت ویرایش و حذف نیز باید در نظر گرفته شود.

مدیر همینطور باید بتواند با مشاهده لیست دوره ها، لیست کلیه افرادی (استاد/دانشجو) که در یک دوره خاص حضور دارند را نیز در صفحه ای جداگانه مشاهده کند.

◀ فاز B

✎ تعریف آزمون توسط استاد

کاربر با نقش استاد، بعد از ورود به وب اپلیکیشن، باید بتواند لیست دوره هایی که در آن ها تدریس می کند و توسط مدیر به آنها اضافه شده است، را مشاهده کند. استاد، با انتخاب هر دوره، باید بتواند لیست آزمون های آن دوره را مشاهده کند.

با ورود به هر دوره نیز، باید لیستی از کلیه آزمون هایی که خودش قبلاً برای آن دوره ایجاد کرده را مشاهده کند. همچنین می تواند آنها را ویرایش کند و یا آزمونی را حذف کند. همینطور در این قسمت می تواند یک آزمون جدید را ایجاد کند.

با انتخاب "افزودن آزمون جدید"، و بعد با وارد کردن "عنوان" برای آزمون، "توضیحات" درباره آزمون و نیز "زمان" مورد نیاز برای آن آزمون بر اساس دقیقه، و در پایان، انتخاب "ذخیره آزمون"، می تواند یک آزمون جدید ایجاد کند. پس از آنکه آزمون با موفقیت ساخته شد، لیست آزمون های دوره، باید آپدیت شود.

◀ فاز C

✎ افزودن سوالات چند گزینه ای و تشریحی به آزمون

استاد، پس از انتخاب "ویرایش آزمون"، اکنون می تواند به دو روش سوالات را به آن آزمون، اضافه کند:

- استفاده از بانک سوالات: در این بخش، کلیه سوالاتی که قبلاً توسط آن استاد، طراحی شده است، قابل مشاهده هست و استاد می تواند سوال یا سوالاتی را از آنها به آن آزمون اضافه کند.
- افزودن یک سوال جدید به آزمون

هدف از این قسمت، طراحی بخشی است که کاربر "استاد" بتواند بعد از انتخاب "افزودن سوال"، نوع سوال را انتخاب کند.

دو نوع سوال وجود دارد:

- سوال چند گزینه ای: استاد، باید بتواند با زدن بر روی دکمه ی "افزودن گزینه ی جدید" به هر تعداد که می خواهد گزینه به سوال مربوطه اضافه کند. نباید محدودیتی در تعداد گزینه ها وجود داشته باشد، و باید جواب صحیح نیز مشخص شود. (تعداد گزینه های یک سوال از قبل مشخص نمی شود)
- سوال تشریحی: در این حالت، متنی به عنوان صورت سوال در نظر گرفته خواهد شد، و مکانی نیز برای پاسخگویی برای دانشجو.

هر سوال، علاوه بر صورت سوال، دارای یک "عنوان" نیز می باشد و به عنوان فیلد اجباری باید در نظر گرفته شود. از این عنوان، برای راحت تر پیدا کردن سوالات در آینده، در بانک سوالات و نیز موقع تعیین نمره، استفاده خواهد شد، و طبیعتاً باید کوتاه باشد!

قابل ذکر است که با افزودن هر سوال جدید، طبیعتاً این سوال باید به صورت اتوماتیک به بانک سوالات همان استاد نیز اضافه شود

⚠ توجه!

- طراحی باید به گونه ای باشد که در آینده، بدون تحت تاثیر قرار گرفتن سایر قسمت های پروژه، انعطاف پذیری در قابلیت اضافه کردن "نوع جدید سوال" نیز، وجود داشته باشد. (اصول طراحی بهینه و SOLID حتما رعایت شود).
- قابلیت ویرایش برای هر سوال نیز وجود داشته باشد.

📌 بانک سوالات

بانک سوالات باید دربرگیرنده کلیه سوالاتی باشد که یک استاد برای تمام آزمون های یک دوره طراحی کرده است. در هر دوره، هر سوال جدیدی که در هر آزمونی از آن، ایجاد می شود باید به طور اتوماتیک به بانک سوالات همان دوره نیز اضافه شود و به عبارت دیگر در کلیه آزمون های همان دوره قابل دسترس باشد.

📌 تعیین نمره پیشفرض برای هر سوال در آزمون

استاد باید با مشاهده لیست سوالات آزمون، بتواند برای هر سوال نمره ای پیشفرض تعیین کند. در پایان نیز باید جمع کل این نمرات برای این آزمون محاسبه و نمایش داده شود، که نشان دهنده **سقف** کل نمره ای خواهد بود که به این آزمون تعلق می گیرد.

⚠ **توجه!** این نمرات، فقط مربوط به همان آزمون است، چرا که باید دقت کرد ممکن است این سوال در یک آزمون دیگر هم به واسطه ی "بانک سوالات" استفاده شود و در آنجا ممکن است نیاز باشد نمره پیشفرض متفاوتی برایش در نظر گرفت!!! پس به این نکته در طراحی خود، توجه کنید!!!

🔍 فاز D

📌 مشاهده لیست دوره های عضو توسط دانشجو

کاربر با نقشی دانشجو، پس از ورود، باید بتواند لیست دوره هایی که در آنها توسط مدیر، در آن ها ثبت نام شده است را ببیند.

دانشجو، پس از ورود به دوره مورد نظر خود، لیستی از آزمون هایی که **مجاز** به شرکت در آنها است را مشاهده می کند. با انتخاب "**شرکت در آزمون**" می تواند در آن آزمون شرکت کند و **بلافاصله** آزمون شروع می شود.

دقت کنید اگر کاربر قبلا در آزمونی شرکت کرده باشد، دیگر **مجاز** به شرکت در آن آزمون **نمی باشد!!!**

📌 برگزاری آزمون

با شروع آزمون، یک شمارشگر به صورت معکوس باید زمان باقی مانده را محاسبه کند و در هر لحظه نمایش دهد. سوالات باید به دانشجو نمایش داده شود، و با زدن دکمه بعدی، سوال بعدی نمایش داده شود. با رسیدن به سوال آخر، و انتخاب دکمه "**ارسال**" آزمون پایان می یابد.

همینطور باید بتواند پیش از پایان زمان آزمون، به سوالات قبلی برگردد، و در صورت نیاز جواب را تغییر دهد.

توجه شود در صورتی که زمان به پایان برسد، آزمون باید اتوماتیک پایان پذیرد. طبیعتاً فقط سوالاتی که دانشجو جواب داده است ثبت می شود و سایر موارد اگر پاسخی داده نشده است، به عنوان پاسخ داده نشده ثبت می گردد و نمره ای به آنها تعلق نمی گیرد، همینطور نمره منفی وجود ندارد.

ممکن است به هر دلیلی ارتباط دانشجو در حین آزمون با وب سرور قطع شود، بنابراین، تمهیداتی را بیندیشید که سوالاتی را که جواب داده است، تا پیش از پایان آزمون، جوابشان در جایی به طور موقت ذخیره شود و با برقراری ارتباط مجدد در همان بازه زمانی، بتواند به آزمون ادامه دهد. بدیهی است در صورتی که زمان آزمون تمام شده باشد، و کاربر مجدد به صفحه اش برگشت دیگر نباید قادر به ادامه شرکت در آزمون باشد!

◀ فاز E

📌 نمره دهی به آزمون:

استاد باید بتواند با بروزرسانی صفحه خود، تعداد و نام افراد شرکت کننده در یک آزمون و نتیجه ی آزمون دانشجویانی که آزمونشان پایان پذیرفته است، را مشاهده کند.

برای آزمون هایی که سوالات چند گزینه ای دارند نمره باید به صورت اتوماتیک در پایان آزمون محاسبه شود و برای سوالات تشریحی آن آزمون، استاد باید بتواند با انتخاب صفحه آزمون هر دانشجو، نمره مورد نظر را برای جواب مربوطه وارد کند.

بدیهی است که نمره ای که استاد برای یک جواب تشریحی می دهد، نباید از سقف نمره در نظر گرفته شده برای آن سوال بیشتر باشد!

⚠ توجه!

- برای پیاده سازی باید از **Java** و **Spring** استفاده شود. محدودیتی در استفاده از تکنولوژی های سمت **front-end** وجود ندارد.
- برای سهولت در انجام پروژه، می توانید از پایگاه داده **H2** استفاده کنید.
- نحوه پیاده سازی **UI** کاملاً بر عهده کاروند و مطابق با سلیقه وی است، اما توجه شود باید این طراحی بهینه و **User-friendly** باشد و کلیه **feature** های خواسته شده در آن مدنظر گرفته شده باشد و به درستی کار کنند.
- قابلیت تعریف عکس برای هر سوال آزمون، نمره امتیازی دارد.
- خطاهای متناظر برای هر سرویس به صورت دقیق باشد. برای مثال در صورت عدم احراز هویت کاربران کد خطای متناظر نمایش داده شود.
- استفاده از یک **Rich Editor** با پشتیبانی از زبان فارسی و قابلیت های راست چین و چپ چین (**RTL/LTR**) نمره امتیازی دارد و مزیت محسوب می شود.
- پیاده سازی **feature** های کاربردی و مفید دیگر به پروژه، نمره امتیازی خواهد داشت و مزیت محسوب می شود.
- استفاده از تکنیک های **i18n and localization** مزیت محسوب می شود.

علی نوری

اسفند ۱۴۰۳؛ مکتب شریف

موفق باشید