

پروژه درس برنامه نویسی پیشرفته

بخش اول

زمستان ۱۴۰۰ - دانشکده علوم ریاضی

دانشگاه صنعتی شریف

سیستم آموزش

شناسه کاربر:

رمز عبور:

6870

متن بالا را وارد کنید:

ورود به سیستم

دریافت فونت

استاد درس:

دکتر مجتبی تفاق

تیم طراحی:

ایمان قدیمی، محمد مهدی زارع، بنیامین بیضایی، علی توسلی

معرفی پروژه

در این پروژه شما قرار است سامانه‌ای مشابه edu را راه‌اندازی کنید.

اهداف بخش اول پروژه

- شناخت پیدا کردن نسبت به توسعه نرم‌افزار (Software Development)
- آشنایی و طراحی رابط‌های کاربری گرافیکی
- آشنایی با معماری‌های طراحی نرم‌افزارهای کاربردی، برنامه‌نویسی شی‌گرا و «تمیز» کد زدن
- شناخت ویژگی‌های اختصاصی زبان جاوا
- کار با فایل‌ها
- ذخیره‌سازی صحیح اطلاعات

رفع اشکال

برای این بخش از پروژه یک جلسه توجیهی برگزار می‌شود که زمان آن متعاقبا اعلام خواهد شد. قبل از جلسه توجیهی، حتما داک را مطالعه کنید. همچنین شما می‌توانید برای رفع پرسش‌ها، اشکال‌ها و ابهام‌های خود از طریق گروه تلگرامی و کوئرا با تیم طراحان پروژه در ارتباط باشید.

نکات مهم

- ممکن است در قسمت‌هایی از داک ابهاماتی وجود داشته باشد، این موارد به وسیله کامنت در همین داک جواب داده می‌شوند و قابل مشاهده هستند.
- در صورتی که درباره جزئیات قسمتی از پروژه توضیح داده نشده، می‌توانید از **خلاصیت** خود استفاده کنید و به روش دلخواه آن را پیاده‌سازی کنید.

ارزیابی و روش تحویل

- این بخش به صورت مجزا از بخش های دیگر ارزیابی و به صورت آنلاین تحویل گرفته می شود.
- هر بخش پروژه باید حداکثر تا تاریخی که به شما اعلام شده است در کوئرا آپلود شود. کدی که در زمان تحویل مورد ارزیابی قرار می گیرد، کد آپلود شده در کوئرا است. دقت کنید که ددلاین های اعلام شده قابل تغییر نیست پس برنامه ریزی لازم را برای رساندن پروژه خود به ددلاین ها داشته باشید.
- دقت کنید که فقط اجرا شدن صحیح کد مدنظر نیست و از شما انتظار می رود که به صورت اصولی و با استفاده از مفاهیم تدریس شده در کلاس کد بنویسید.
- بخش هایی که با رنگ سبز نشان داده شده اند امتیازی هستند.

موارد غیر مجاز

- عدم تسلط کافی بر کد پروژه
 - شباهت بیش از حد دو یا چند پروژه
 - واگذاری کامل یا بخشی از پروژه به شخصی دیگر
- رخ دادن این اتفاق ها برای هیچ فردی قابل پذیرش نیست و در صورت بروز هر کدام از این اتفاق ها ممکن است هر تصمیمی در رابطه با ارزیابی فرد گرفته شود.
- در صورتی که یکی از این اتفاق ها رخ داده باشد لازم است که افراد حتما دلیل این مساله را پیش از تشخیص توسط تیم درس اعلام کنند و دلیل این اتفاق را توضیح دهند. در این صورت فقط ارزیابی مربوط به بخش اعلام شده تحت تاثیر قرار خواهد گرفت.
- در صورتی که مشاهده یکی از این اتفاق ها توسط تیم درس، پیش از اعلام فرد رخ دهد این تیم از فرد درخواست خواهد کرد که در این رابطه توضیح دهند و در صورت قابل قبول نبودن توضیح، فرد موردنظر موفق به گذراندن درس نخواهد شد.

تعریف پروژه

- هر چند همه ما با این سامانه آشنا هستیم، اما در این بخش یک دید کلی از اشیا موجود در پروژه می‌دهیم.
- در این پروژه، کاربرانی مثل دانشجو، استاد و معاون آموزشی وجود دارند. همه ما از صفحات و امکاناتی که دانشجو در اختیار دارد اطلاع داریم و می‌توانیم آن‌ها را در سامانه edu انجام دهیم، اما در این پروژه بخش‌هایی نیز اضافه می‌شوند که مربوط به اساتید هستند و دانشجو به این موارد دسترسی ندارد. مثل دانشجویان یک درس، نمرات آن‌ها، تعریف تکلیف و آزمون و... بعلاوه معاون آموزشی هم یک نوع کاربر است که دسترسی‌ها و امکانات متفاوتی از استاد و دانشجو دارد. در این پروژه تمرکز ما بر کاربر دانشجو است، اما از کاربران دیگر هم بخش‌های کمی را پیاده خواهیم کرد.
 - یک موجودیت مهم دیگر در این پروژه، دروس می‌باشند. دروس شامل اطلاعات مختلفی مانند نام، شماره درس، دانشکده، پیشنیازها، همنیازها و... دارند.
 - بعلاوه در این پروژه، سامانه‌ای شبیه به CW نیز پیاده می‌شود. در این سامانه کارهایی شبیه برگزاری آزمون توسط استاد، تحویل آزمون توسط دانشجو، برگزاری تکلیف و تحویل آن توسط دانشجو و... می‌باشد.
 - دقت کنید موارد گفته شده تنها بخش‌هایی از پروژه هستند و ممکن است شما به تعریف کلاس‌های دیگر که ذکر نشده نیاز پیدا کنید. در مجموع معماری این پروژه دست خود شماست و باید سعی کنید تا حد امکان از اصول برنامه‌نویسی شی‌گرا و مزایای آن استفاده کنید.
 - در فاز بعدی ممکن است بخش‌های دیگری اضافه شوند. پس سعی کنید از معماری خوب و قابل گسترشی برای پروژه استفاده کنید.
 - یکی از اهداف اصلی این درس و پروژه، ارتقای و شی‌گرایی هست. تمام تلاش خود را کنید که از آن به خوبی استفاده کنید.
 - در این پروژه تمام نقش‌های موجود کاربران عبارت‌اند از: ۱. دانشجو (کارشناسی - ارشد - دکتری) ۲. استاد ۳. استاد - معاون آموزشی ۴. استاد - رئیس دانشکده
 - هر دانشکده فقط یک رئیس دانشکده دارد.

- هر دانشکده **حداکثر** یک معاون آموزشی دارد (رئیس دانشکده نمی‌تواند معاون آموزشی نیز باشد).

گرافیک

همه ما می‌دانیم که یکی از بخش‌های مهم هر اپلیکیشن بخشی است که کاربر با آن سروکار دارد و قادر است خواسته‌های خود را به برنامه بگوید و پاسخ خود را بگیرد! شما در حین پیاده‌سازی این پروژه باید یک User Interface ساده طراحی و پیاده‌سازی کنید که کاربر با استفاده از آن بتواند توابع مختلفی از برنامه را صدا بزند، تغییرات مورد نظر خود را اعمال کند و پاسخ مناسبی دریافت کند.

در دنیای مدرن امروزی کمتر برنامه‌ای یافت می‌شود که رابط کاربری آن با زبان جاوا و بدون استفاده از کتابخانه‌های گرافیکی مخصوص و یا ابزارهای توسعه پیشرفته رابط کاربری نوشته شده باشد اما شما در این پروژه به اقتضای این‌که مجاز نیستید با زبان برنامه‌نویسی دیگری کار کنید رابط کاربری خود را باید با زبان جاوا بنویسید. مجاز هستید از کتابخانه‌های توسعه گرافیکی مانند Swing و JavaFX استفاده کنید و یا می‌توانید با هر روش دیگری رابط گرافیکی خود را با استفاده از زبان جاوا پیاده‌سازی کنید.

توصیه می‌شود تا حد ممکن سعی کنید که رابط گرافیکی برنامه از منطق برنامه جدا باشد این موضوع را نه تنها در این پروژه بلکه به طور معمول باید در مهندسی همه نرم‌افزارها رعایت کنید تا به مشکل برخورد.

بهتر است یک سیستم بر اساس Request و Response طراحی کنید. رابط گرافیکی شما درخواستی ایجاد کرده و به منطق برنامه شما می‌فرستد، منطق برنامه شما در پشت سر این درخواست را بررسی می‌کند و پاسخ مناسب درخواست را به رابط کاربری پس می‌دهد و رابط کاربری با توجه به پاسخی که دریافت کرده است خود را آپدیت می‌کند. تمامی داده‌های برنامه، محاسبات پیچیده برنامه، سطح دسترسی‌ها، ارث‌بری کلاس‌ها و... مربوط به منطق برنامه و تمامی اشیای گرافیکی مربوط به GUI هستند.

لینک زیر، داکيومنت‌های اصلی جاوا تحت لیسانس Oracle است که رابط گرافیکی جاوا و Swing را توضیح داده است. این توضیحات بسیار جامع و کامل هستند و در اکثر مواقع برای آشنایی با کلاس‌ها و تابع‌های آن‌ها مفید و کافی هستند:

[Link1](#)

این سایت یکی از بهترین منابع آموزشی برای جاوا و به طور خاص برای گرافیک است:

[Link1](#)

لاگ کردن (Logging)

یکی از اصول مهمی که در نوشتن هر برنامه‌ی با کیفیتی رعایت می‌شود، لاگ کردن رخدادهای برنامه است به طوری که بتوان با خواندن لاگ، به روندهای اجرا شده و تغییرات انجام شده پی‌برد. این قابلیت باعث می‌شود نگهداری و توسعه کد با سرعت بسیار بیشتری انجام شود.

در این پروژه از شما انتظار می‌رود این اصل را در تمامی فازها رعایت کنید و لاگ‌های ساختارمندی ذخیره کنید. به این منظور پیشنهاد می‌کنیم برای تولید لاگ، از کتابخانه‌های موجود استفاده کنید. این کتابخانه‌ها، مسئولیت ذخیره لاگ، حذف لاگ‌های قدیمی، تولید لاگ به فرمت مشخص (سطح اهمیت لاگ، زمان، کلاس تولید کننده لاگ و...)، ارتباط با دیتابیس و... را بر عهده دارند. اما می‌توانید خود یک ابزار ساده لاگ کردن هم توسعه بدهید.

چگونه لاگ کنیم؟

دقت کنید که هدف ما از لاگ کردن، متوجه شدن روند کلی برنامه و اتفاقات رخ داده است و همچنین نباید لاگ کردن، سربار عملیاتی زیادی داشته باشد. برای مثال به هنگام برخورد کردن به خطای زمان اجرا، ذخیره کردن Stack Trace کار خوبی نیست. گرچه اطلاعات زیادی در اختیار ما قرار می‌دهد ولی بسیار حجیم است و برای درک آن، نیازمند درک کد هستیم. به علاوه، در صورتی که از اتفاقات پیش از رخداد خطا بی‌خبر باشیم، ممکن است درک خطا را دشوار کند. بنابراین به دنبال داشتن یک خلاصه از اجرای روند برنامه هستیم.

اصول خوبی از لاگ کردن را می‌توانید در [این صفحه](#) ببینید. نمره‌دهی به طور کلی بر اساس موارد داخل این صفحه خواهد بود. موارد ۱، ۲، ۴، ۶ و ۷ مهم‌ترین مواردی هستند که بهتر است رعایت کنید.

تعدادی از مواردی که می‌توانید در لاگ رعایت کنید:

- چه کلاسی
- چه تابعی
- چه زمانی
- روی چه داده‌هایی

- در چه شرایطی

- چه نتیجه‌ای

چه چیز را لاگ کنیم؟

از شما انتظار می‌رود مشابه موارد زیر را لاگ کنید:

- شروع برنامه

- باز کردن و بستن فایل

- ورود یا ثبت نام کاربر

- تغییر اطلاعات کاربری

- تغییر یا افزودن اطلاعات

- خطاهای مربوط به منطق برنامه

- خطاهای معمول زمان اجرا (برای مثال عدم توانایی ایجاد فایل و...)

صفحه ورود

در این قسمت باید استاد، دانشجو و معاون آموزشی بتوانند وارد اکانت‌های خود شوند. این امر به وسیله شناسه کاربر و رمز عبور صورت می‌گیرد. انتخاب این‌که چه چیزی شناسه کاربر باشد به عهده خودتان است اما بدیهتاً شناسه هر کاربر باید با دیگری فرق کند. این صفحه دارای قابلیت‌های زیر می‌باشد:

☐ کپچا

☐ هش کردن رمز عبور

- کپچا: در صفحه لاگین باید یک کپچا به صورت تصویر (png, jpeg و ...) نمایش داده شود که شامل یک عدد چهار رقمی است. اگر کاربر کپچا را درست وارد کرد، در صورتی که نام کاربری و رمز عبور صحیح بود وارد اکانت خود می‌شود. (طبعاً با توجه به نقش خود، وارد صفحه اصلی مخصوص به خود

می‌شود.) در غیر این صورت باید پیغام خطای مناسب داده شود و کپچا عوض شود. داشتن حداقل پنج کپچای مختلف برای سامانه الزامی است همچنین کاربر باید بتواند کپچای نمایش داده شده را عوض کند.

- رمز عبور نباید به همان صورتی که کاربر وارد کرده در دیتابیس ذخیره شود و باید هش شود (در این فاز دیتابیس شما فایل‌های txt یا json هستند) علت این کار این است که اگر به هر دلیلی دیتابیس لو برود، شخص ثالث نتواند با دیتابیس لو رفته وارد اکانت‌های موجود شود.

نکات:

- نیازی به الگوریتم‌های پیچیده کپچا نیست. می‌توانید تصاویر کپچاها را از سایت edu بردارید و به هر کپچا یک عدد که متناظر با عدد آن است نسبت دهید. سپس کافی است چک کنید عددی که کاربر وارد کرده با عددی که به کپچا نسبت دادید یکسان است یا خیر.
 - نیاز به الگوریتم‌های پیچیده هش نیست. در حالت کلی، الگوریتم‌های هش باید به گونه‌ای باشند که از عبارت هش شده نتوان به عبارت اصلی پی برد. اما شما نیازی به پیاده‌سازی این نوع هش‌ها ندارید. برای راحتی می‌توانید از کتابخانه‌های موجود استفاده کنید و یا یک تابع هش برای خود بنویسید.
- برای کسب اطلاعات بیشتر درباره هش می‌توانید از لینک‌های زیر استفاده کنید:

[Link1](#)

[Link2](#)

فایل اولیه

شما همراه کدها و منابع این فاز، باید یک دیتای اولیه آماده کنید و ارسال کنید. این دیتا در روند تحویل و تست برنامه کمک می‌کند.

انتظار می‌رود که حداقل اشیا زیر در فایل باشند:

- ☐ کاربران سایت (دانشجو (کارشناسی، ارشد، دکتری)، استاد، معاون آموزشی و رئیس دانشکده) هر کدام حداقل دو نفر
- ☐ درس‌های دانشکده‌های مختلف (هر دانشکده حداقل دو درس و حداقل پنج دانشکده)
- ☐ حداقل دو دانشجو درس داشته باشند.

نکات:

- این دیتای اولیه لزومی ندارد در یک فایل باشد و می‌تواند به ازای هر کاربر یا درس فایل جداگانه‌ای ذخیره شود.
- رئیس دانشکده فقط از طریق فایل اولیه قابل ساختن است.
- هدف از این فایل خواندن یک دیتای اولیه است. که در فاز بعدی در سمت سرور با خواندن آن، دیتای کاربران و دروس در اختیار سرور قرار می‌گیرد.
- دقت کنید که کاربرانی که در دیتای اولیه هستند، باید بتوانند وارد سایت شوند و اطلاعات آن‌ها کافی و کامل باشد و با اجرا شدن برنامه این دیتا لود شود.
- توصیه می‌شود از کتابخانه‌های Json, Gson, Jackson استفاده کنید. لینک‌های زیر به شما در یادگیری آن‌ها کمک می‌کند:

[Link1](#)

[Link2](#)

[Link3](#)

صفحه اصلی

پس از صفحه لاگین باید وارد صفحه اصلی شوید (مشابه صفحه اصلی edu). طراحی گرافیکی این صفحه دست خودتان است ولی توجه کنید که کاربران باید بتوانند از این صفحه به همه صفحات دیگر دسترسی داشته باشند و به همین صفحه برگردند.

مواردی که باید نمایش دهید:

برای همه کاربران:

☐ زمان جاری (ساعت و روز)

☐ دکمه خروج

☐ زمان آخرین ورود

☐ عکس کاربر

☐ نام و ایمیل کاربر

از دید دانشجو:

- ☐ وضعیت تحصیلی
- ☐ استاد راهنما
- ☐ مجوز ثبت نام
- ☐ ساعت ثبت نام
- ☐ امور ثبت نام و زیرمنوهای آن
- ☐ خدمات آموزشی و زیرمنوهای آن
- ☐ امور کارنامه و زیرمنوهای آن
- ☐ پروفایل کاربر و زیرمنوهای آن

از دید اساتید:

- ☐ امور ثبت نام و ترمیم
- ☐ خدمات آموزشی
- ☐ کارنامه دروس و دانشجوها
- ☐ پروفایل کاربر

از دید معاون آموزشی:

- ☐ معاون آموزشی علاوه بر اینکه تمام منو اساتید را دارا است یک گزینه برای ثبت کاربر جدید هم دارد.

● ثبت کاربر جدید:

این گزینه معاون آموزشی را قادر می‌کند که بتواند یک استاد یا دانشجوی جدید را به سامانه اضافه کند. معاون آموزشی قادر است تمامی اطلاعات کاربر را وارد کرده و یک نام کاربری و کلمه عبور برای او تخصیص دهد.

نکات:

- کاربر باید بتواند از همه صفحات به صفحه اصلی باز گردد.

- کاربر باید بتواند هر لحظه از برنامه خارج شود.
- هر کدام از موارد بالا که نیاز به توضیح بیشتر دارند در ادامه توضیح داده شده است و حتی خودتان می‌توانید با مراجعه به سایت مرجع از امکانات آن‌ها مطلع شوید.

امور ثبت نام

برای دانشجوها:

☐ لیست دروس

☐ لیست اساتید

- لیست دروس:

در این صفحه باید تمام دروس ترم جاری و اطلاعاتشان را نشان دهید. همچنین دانشجو بتواند با اعمال فیلترهای متفاوت درس‌های مطلوب خود را ببیند. فیلترها می‌توانند شامل: نام دانشکده، تعداد واحد، نام استاد، مقطع درس، شماره درس و... باشد. (حداقل بر اساس سه مشخصه فیلتر کنید.)

- لیست اساتید:

در این قسمت دانشجو باید بتواند مانند لیست دروس، لیست اساتید را ببیند و تعدادی از اطلاعات اساتید قابل دیدن باشد. علاوه بر اسم استاد، حداقل ۳ مورد از: دانشکده و ایمیل و درجه استادی و سمت در دانشگاه و شماره اتاق و تماس را نشان دهید. همچنین قابلیت فیلتر کردن بین اساتید نمره امتیازی دارد. برای این بخش مشابه فیلتر کردن دروس می‌توانید عمل کنید و بر اساس حداقل سه مشخصه نام، درجه استادی و دانشکده فیلتر کنید.

برای اساتید:

☐ لیست دروس

☐ لیست اساتید

● لیست دروس:

مانند دانشجویها است.

● لیست اساتید:

○ این بخش مانند بخش "لیست اساتید" برای دانشجو است.

○ اگر این استاد رئیس دانشکده بود، باید بتواند اساتید دانشکده خود را ویرایش، حذف یا اضافه کند.

○ رئیس دانشکده قادر است نقش اساتید را به معاون آموزشی تغییر دهد. اگر دانشکده دارای

معاون آموزشی باشد رئیس دانشکده قادر به انتصاب معاون آموزشی جدید نیست و باید حتما

معاون آموزشی قبلی را عزل کند.

برای معاون آموزشی:

☐ لیست دروس

☐ لیست اساتید

● لیست دروس:

معاون آموزشی همچنین باید بتواند درس‌های دانشکده خود را ویرایش، اضافه یا حذف کند.

(اساتید و دانشجویها فقط می‌توانند ببینند.)

● لیست اساتید:

این بخش مانند بخش "لیست اساتید" برای دانشجو است.

نکات:

● می‌توانید همه اساتید را استاد راهنما هم قرار دهید، از جمله معاون آموزشی.

● موارد گفته شده بالا حداقل‌ها هستند، به سلیقه خود می‌توانید اطلاعات و امکانات بیشتری قرار دهید.

● نحوه نمایش تماما به عهده و سلیقه خودتان است.

خدمات آموزشی

این منو برای کاربر های متفاوت فرق می کند. هر کدام را توضیح می دهیم ولی نحوه نمایش به عهده خودتان است.
از دید دانشجو کارشناسی

☐ برنامه هفتگی:

☐ لیست امتحانات دانشجو:

☐ درخواست ها:

☐ درخواست توصیه نامه

☐ درخواست گواهی اشتغال به تحصیل

☐ ماینور

☐ انصراف از تحصیل

● برنامه هفتگی:

برنامه هفتگی دانشجو را به شکل دلخواه نمایش دهید.

● لیست امتحانات:

لیست امتحانات دانشجو را به ترتیب زمانی نشان دهید.

● درخواست توصیه نامه:

دانشجو بتواند به استاد مربوطه درخواست توصیه نامه بدهد و نتیجه را هم در همین صفحه ببیند.

اینجانب ... گواهی می دهم که آقا/خانم ... به شماره دانشجویی ...، درس های ... را با نمره ... گذرانده اند و همچنین در دروس ... به عنوان دستیار آموزشی فعالیت داشته اند.

● درخواست گواهی اشتغال به تحصیل:

دانشجو در این منو به صورت رایگان درخواست گواهی بدهد و در اختیار او قرار بگیرد. از شما انتظار نمی‌رود که یک گواهی برای او صادر کنید. می‌توانید یک متن از قبل آماده شده در یک فایل داشته باشد و آن را با توجه به مشخصات دانشجو تکمیل کنید و در این صفحه نمایش دهید. به عنوان مثال متن زیر کافی است:

گواهی می‌شود که آقا خانم ... به شماره دانشجویی ...، مشغول به تحصیل در رشته ... در دانشگاه ... است. تاریخ اعتبار گواهی: ...

● ماینور:

یک سری شروط مانند معدل برای قبولی درخواست به انتخاب خود قرار دهید. اگر دانشجو حائز این شرایط بود درخواست او را به سمت معاون آموزشی دانشکده مبدا و مقصد بفرستید به شکلی که معاون آموزشی قادر باشد این درخواست را قبول یا رد کند و دانشجو وضعیت درخواست خود را ببیند. درخواست دانشجو سه وضعیت کلی خواهد داشت: ۱. ثبت شده ۲. تایید شده (دانشکده مبدا و مقصد قبول کرده باشند). ۳. رد شده (حداقل یکی از دانشکده‌های مبدا و مقصد رد کرده باشند).

● انصراف از تحصیل:

دانشجو بتواند از این طریق، از تحصیل انصراف دهد. این درخواست نیز به سمت معاون آموزشی می‌رود و معاون قادر است درخواست انصراف را رد یا قبول کند در صورت این‌که با انصراف دانشجو موافقت شود دانشجو دیگر نمی‌تواند وارد سامانه شود و همچنین وضعیت تحصیل وی عوض می‌شود.

از دید دانشجو کارشناسی ارشد:

☐ برنامه هفتگی

☐ لیست امتحانات دانشجو

☐ درخواست‌ها:

☐ درخواست توصیه‌نامه

☐ درخواست گواهی اشتغال به تحصیل

☐ درخواست خوابگاه

☐ انصراف از تحصیل

● برنامه هفتگی:

برنامه هفتگی دانشجو را به شکل دلخواه نمایش دهید.

● لیست امتحانات:

لیست امتحانات دانشجو را به ترتیب زمانی نشان دهید.

● درخواست توصیه‌نامه:

دانشجو بتواند به استاد مربوطه درخواست توصیه‌نامه بدهد و نتیجه را هم در همین صفحه ببیند.

اینجانب ... گواهی می‌دهم که آقا/خانم ... به شماره دانشجویی ...، درس‌های ... را با نمره ... گذرانده‌اند و همچنین در دروس ... به عنوان دستیار آموزشی فعالیت داشته‌اند.

● درخواست گواهی اشتغال به تحصیل:

دانشجو در این منو به صورت رایگان درخواست گواهی بدهد و در اختیار او قرار بگیرد. از شما انتظار نمی‌رود که یک گواهی برای او صادر کنید. می‌توانید یک متن از قبل آماده شده در یک فایل داشته باشد و آن را با توجه به مشخصات دانشجو تکمیل کنید و در این صفحه نمایش دهید. به عنوان مثال متن زیر کافی است:

گواهی می‌شود که آقا/خانم ... به شماره دانشجویی ...، مشغول به تحصیل در رشته ... در دانشگاه ... است. تاریخ اعتبار گواهی: ...

● درخواست خوابگاه:

دانشجو بتواند درخواست خود را ثبت کند و به صورت تصادفی رد یا پذیرفته شود.

● انصراف از تحصیل:

مانند دانشجوی کارشناسی

از دید دانشجو دکتری:

☐ برنامه هفتگی

☐ لیست امتحانات دانشجو

☐ درخواست ها:

☐ درخواست گواهی اشتغال به تحصیل

☐ انصراف از تحصیل

☐ درخواست نوبت دفاع از پایان نامه

● برنامه هفتگی:

برنامه هفتگی دانشجو را به شکل دلخواه نمایش دهید.

● لیست امتحانات:

لیست امتحانات دانشجو را به ترتیب زمانی نشان دهید.

● درخواست گواهی اشتغال به تحصیل:

مانند دانشجوی کارشناسی

● انصراف از تحصیل:

مانند دانشجوی کارشناسی

● درخواست نوبت دفاع از پایان نامه:

دانشجو درخواست نوبت دفاع از پایان نامه بدهد و برای او یک نوبت ثبت شود. این نوبت

ثبت شده را می‌توانید برای او به شکل روبه‌رو نشان دهید: در تاریخ فلان می‌توانید از

پایان نامه خود دفاع کنید.

امور کارنامه

نامه شامل دو بخش کلی زیر است:

☐ نمرات موقت

☐ وضعیت تحصیلی دانشجو

■ نمرات موقت:

برای دانشجو:

- دانشجو باید بتواند نمراتی که به طور موقت ثبت شده‌اند را مشاهده کند.
- دانشجو قادر است برای این نمرات اعتراضی ثبت کند.
- دانشجو قادر است جوابیه استاد را ببیند.
- وقتی نمره از حالت موقت خارج می‌شود و به حالت ثبت شده در می‌آید دانشجو دیگر نمی‌تواند نمره را در لیست نمرات موقت ببیند.

برای استاد:

- استاد لیست دانشجویهای ثبت‌نامی درس را مشاهده کرده و برای هر کدام می‌تواند نمره جداگانه وارد کند.
- استاد قادر است قبل از ثبت نمرات نمره هر دانشجو را ویرایش کند.
- استاد قادر نیست نمرات موقت را ثبت کند مگر اینکه یک نمره موقت برای همه وارد کرده باشد.
- استاد قادر نیست بیشتر و یا کمتر از رنج ۰ تا ۲۰ برای دانشجو نمره‌ای در نظر بگیرد و در غیر این صورت خطای مناسب باید به او نشان داده شود.
- چنانچه نمره وارد شده استاد دقتی بیشتر از ۰.۲۵ داشته باشد سیستم به طور خودکار نمره را گرد می‌کند.
- استاد قادر است اعتراضات دانشجویها را مشاهده کند.
- استاد قادر است یک جواب برای هر اعتراض بنویسد و نمره دانشجو را ویرایش کند.
- استاد قادر است نمرات را ثبت نهایی کند. (بعد از این که ثبت موقت کرد.)

برای معاون آموزشی:

- معاون آموزشی قادر است نمرات ثبت شده همه دانشجویان یک کلاس را مشاهده کند.
- معاون آموزشی قادر است نمرات موقت یک دانشجو خاص را فیلتر کند و ببینید.
- معاون آموزشی قادر است بر اساس نام استاد جستجو کند و نمرات ثبت شده توسط وی را مشاهده کند.
- معاون آموزشی قادر است اعتراض‌ها و پاسخ اعتراض‌ها را مشاهده کند.
- معاون آموزشی قادر است یک خلاصه از درس را مشاهده کند. این خلاصه شامل معدل درس، تعداد افراد مردود و قبول شده و معدل درس بدون در نظر گرفتن افراد مردود است.

■ وضعیت تحصیلی دانشجو:

این بخش فقط برای دانشجو و معاون آموزش پیاده‌سازی می‌شود:

برای دانشجو:

- دانشجو قادر است تعداد واحدهایی را که پاس کرده است مشاهده کند.
- دانشجو قادر است نمراتی که ثبت نهایی شده‌اند را در این بخش مشاهده کند. (درس‌هایی که نمره آنها هنوز ثبت نهایی نشده‌اند را می‌توانید به صورت N/A نمایش دهید.)
- دانشجو معدل خود را که به صورت وزن‌دار محاسبه می‌شود را در این بخش مشاهده می‌کند.

برای معاون آموزشی:

- معاون آموزش قادر است تمامی مواردی که دانشجو مشاهده می‌کند را مشاهده کند.
- معاون آموزشی قادر است بر اساس اسم دانشجو و شماره دانشجویی در بین داده‌های موجود جستجو کند.

پرو فایل کاربر

برای دانشجو:

□ نام و نام خانوادگی

- ☐ کد ملی
- ☐ شماره دانشجویی
- ☐ شماره موبایل
- ☐ پست الکترونیک
- ☐ معدل کل
- ☐ دانشکده
- ☐ استاد راهنما
- ☐ سال ورود
- ☐ عکس دانشجو
- ☐ مقطع (کارشناسی، کارشناسی ارشد، دکتری)
- ☐ وضعیت (در حال تحصیل، فارغ التحصیل یا انصراف از تحصیل)

برای استاد:

- ☐ نام و نام خانوادگی
- ☐ کد ملی
- ☐ شماره استادی
- ☐ شماره موبایل
- ☐ پست الکترونیک
- ☐ دانشکده
- ☐ شماره اتاق
- ☐ عکس استاد
- ☐ درجه استادی (شامل سه درجه: استادیار، دانشیار، استاد تمام)

برای همه کاربران:

□ هر کاربر بتواند تم را عوض کند.

نکات:

- توجه کنید که هر دو نوع کاربر (دانشجو، استاد) باید بتوانند پست الکترونیک و شماره موبایل خود را تغییر دهند.
- قسمت عکس استاد و دانشجو در صورتی که عکسی از آن‌ها موجود نباشد باید خالی باشد. (برای مثال یک عکس پیش‌فرض برای کاربرانی که عکس ندارند نمایش داده شود.)
- نحوه عوض کردن تم به عهده خودتان است. (می‌تواند صرفاً تغییر رنگ‌ها باشد.)

صفحه تغییر رمز عبور

□ اگر بیش از سه ساعت از آخرین عبور کاربری گذشته باشد، یک صفحه تغییر رمز نمایان می‌شود که کاربر مجبور است رمز عبور خود را تغییر دهد.

نکات:

- در این صفحه رمز قبلی کاربر و رمز جدیدش خواسته می‌شود.
- تا زمانی که کاربر با موفقیت از این صفحه عبور نکند، اجازه ورود به او داده نمی‌شود.

ضمن تبریک سال نو و آرزوی موفقیت

تیم درس برنامه نویسی پیشرفته

