

UMIL مدرسه

آموزشگاه فنی حرفه ای

میناب

حمیدرضا خستو

علیرضا زارعی

توضیح چهارده نمودار (UML 2 (Unified Modeling Language

- **نمودار کلاسی (Class Diagram)**
- **اجزا:** کلاس‌ها، ویژگی‌ها، عملیات‌ها، روابط و وابستگی‌ها.
- **کاربرد:** نمایش ساختار ایستای سیستم از طریق نمایش کلاس‌های سیستم و روابط بین آن‌ها.
- **مزایا:** وضوح درک ساختار داده‌ها و روابط میان آن‌ها.
- **معایب:** می‌تواند پیچیده شود اگر سیستم بزرگ باشد.
- **جنبه‌های ایستایی:** ساختار و سازماندهی داده‌ها.
- **جنبه‌های پویایی:** ندارد.

یک دیاگرام کلاس UML ساده را نشان میدهد که سه کلاس اصلی دارد: Student دانشجو، Teacher استاد، و Course دوره. هر کلاس شامل فیلدها (ویژگیها) و متدها (عملیات) است. توضیحات جزئیتر به شرح زیر است:

1. کلاس: Student دانشجو

فیلدها (خصوصیات):

studentId: String شماره دانشجویی

name: String نام دانشجو

متدها (عملیاتها):

registerCourse(course: Course): void برای ثبتنام دانشجو در یک دوره استفاده میشود.

getGrades(): List<Grade>: لیستی از نمرات دانشجو را برمیگرداند (احتمالاً کلاس Grade بهصورت جداگانه تعریف شده است).

2. کلاس: Teacher استاد

- فیلدها:

teacherId: String شماره استاد

name: String نام استاد

- متدها:

assignGrade(student: Student, grade: Grade): void به استاد اجازه میدهد به یک دانشجو نمره اختصاص دهد.

3. کلاس: Course دوره

- فیلدها:

courseCode: String کد دوره

title: String عنوان دوره

- متدها:

getStudents(): List<Student> لیست دانشجویان ثبتنامشده در این دوره را برمیگرداند.

نکات مهم و ارتباط بین کلاسها

1. کلاس Grade در دیاگرام نشان داده نشده است، اما وجود آن از طریق متدهای getGrades و assignGrade قابل استنباط است. احتمالاً این کلاس شامل اطلاعاتی مانند نمره، درس مرتبط، و تاریخ اختصاص نمره است.

2. ارتباطها:

یک Student میتواند در چندین Course ثبتنام کند (registerCourse).

یک Teacher میتواند به چندین Student نمره اختصاص دهد (assignGrade).

یک Course میتواند لیست دانشجویان خود را مدیریت کند (getStudents).

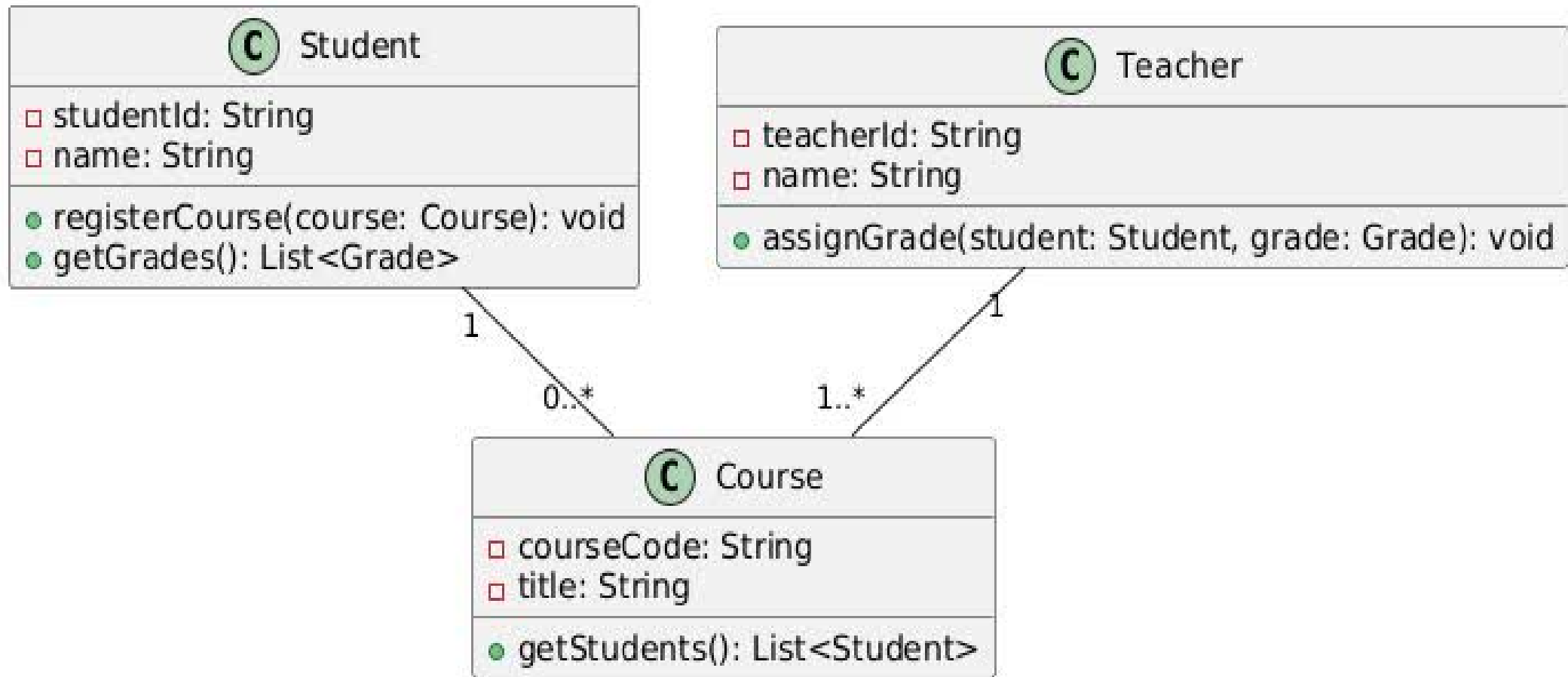
3. سطح دسترسی:

نماد □ نشاندهنده فیلدهای خصوصی (private) است.

نماد ● نشاندهنده متدهای عمومی (public) است.

کاربرد دیاگرام

این دیاگرام ساختار اصلی یک سیستم مدیریت آموزشی را نشان میدهد که در آن دانشجویان در دورهها ثبتنام میکنند، اساتید نمرات را وارد میکنند، و دورهها اطلاعات دانشجویان را مدیریت میکنند. با این حال، جزئیات برخی کلاسها (مانند Grade) یا ارتباطهای پیچیدهتر (مانند ارثیری) در این نسخه سادهشده وجود ندارد.



➤ نمودار مورد کاربرد (Use Case Diagram)

- اجزا: بازیگران، موارد کاربرد، روابط.
- کاربرد: نمایش نیازمندی‌های عملکردی سیستم.
- مزایا: ساده و قابل فهم برای کاربران غیر فنی.
- معایب: جزئیات فنی را نشان نمی‌دهد.
- جنبه‌های ایستایی: ندارد.
- جنبه‌های پویایی: تعاملات بین کاربران و سیستم.

یک نمودار فرآیند یا فلوچارت ساده برای سیستم مدرسه را نشان می‌دهد که مراحل و اجزای مرتبط با ثبت‌نام دانشجو در دوره‌های آموزشی و مشاهده نمرات را توصیف میکند. اجزای اصلی و توضیحات آنها به شرح زیر است:

1. عنوان اصلی: سیستم مدرسه

این بخش به کلیت سیستم مدیریت مدرسه اشاره دارد که احتمالاً شامل فرآیندهای مختلفی مانند ثبت‌نام، مدیریت نمرات، و تأییدیه‌ها است.

2. بخش مشاهده نمرات

- این قسمت شامل مراحل یا اجزای مرتبط با مدیریت نمرات دانشجویان است:
- Student دانشجو: نقطه شروع فرآیند، جایی که دانشجو به سیستم وارد میشود.
 - ثبت نام در دوره: دانشجو درخواست ثبتنام در یک دوره آموزشی را میدهد.
 - include: احتمالاً به معنای "شامل شدن" است، مانند افزودن دانشجو به لیست دوره یا اتصال به سیستم نمرات.
 - تایید ثبت نام: مرحله نهایی که ثبتنام دانشجو توسط مسئول مربوطه (مانند دفتر ثبت) تأیید میشود.

3. Registrar دفتر ثبت

- این بخش در پایین نمودار قرار دارد و نشاندهنده نهاد مسئول تأیید ثبتنامها است. دفتر ثبت احتمالاً:
- درخواستهای ثبتنام دانشجویان را بررسی میکند.
 - پس از تأیید، دانشجو را بهطور رسمی در دوره ثبتنام میکند.
 - ممکن است به بخش مشاهده نمرات دسترسی داشته باشد تا نمرات نهایی را وارد یا ویرایش کند.

نکات کلیدی و ارتباطات

1. فرآیند ثبتنام:

- دانشجو ابتدا درخواست ثبتنام میدهد (ثبت نام در دوره).
- سیستم دانشجو را به دوره اضافه میکند
- دفتر ثبت (Registrar) درخواست را بررسی و تأیید میکند (تایید ثبت نام).

2. مشاهده نمرات:

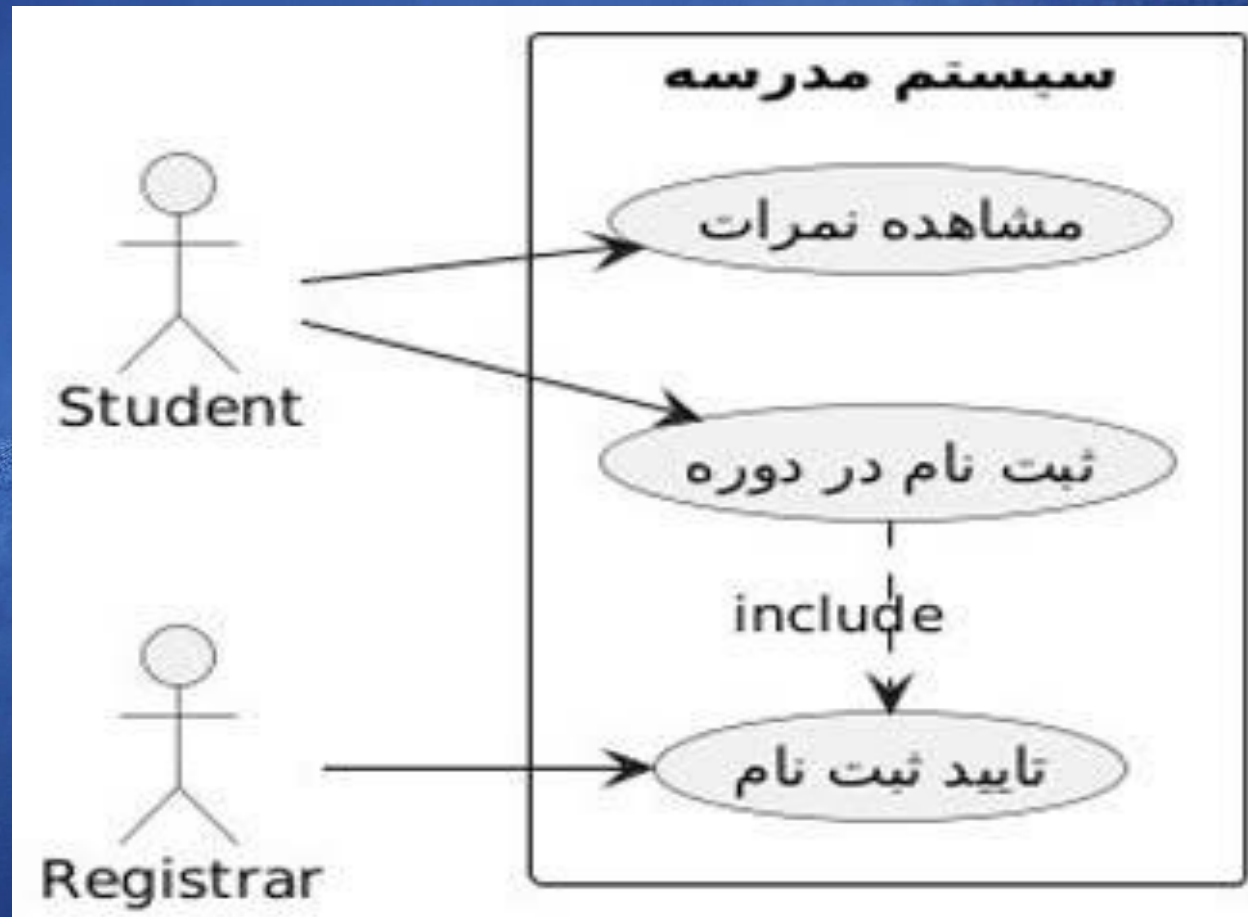
- پس از ثبتنام، احتمالاً نمرات دانشجو در این بخش قابل مشاهده یا ویرایش است.
- ارتباط بین Student و Registrar نشاندهنده نقش دفتر ثبت در مدیریت اطلاعات دانشجویی است.

3. عدم وضوح برخی جزئیات:

- کلاسها یا موجودیتهایی مانند Course یا Grade در این نمودار نشان داده نشدهاند.
- عبارت include ممکن است نیاز به توضیح بیشتری داشته باشد (مثلاً شامل شدن در لیست دوره، پایگاه داده، یا سیستم نمرات).

کاربرد نمودار

این نمودار یک نمای ساده از گردش کار ثبتنام دانشجو و مدیریت نمرات در یک سیستم مدرسه ارائه میدهد. با این حال، جزئیات فنی (مانند نوع دادهها، روابط پیچیده، یا مکانیسمهای احراز هویت) در آن وجود ندارد و احتمالاً برای درک کلی فرآیند طراحی شده است.



➤ نمودار بسته (Package Diagram)

- اجزا: بسته‌ها، وابستگی‌ها.
- کاربرد: سازماندهی کلاس‌ها به بسته‌های منطقی.
- مزایا: سازماندهی بهتر کلاس‌ها.
- معایب: ممکن است برای سیستم‌های کوچک لازم نباشد.
- جنبه‌های ایستایی: سازماندهی کلاس‌ها.
- جنبه‌های پویایی: ندارد.

این تصویر یک دیاگرام ماژولار یا ساختار کلی سیستم مدیریت آموزشی را نشان می‌دهد که سه بخش اصلی دارد: مدیریت کاربران، مدیریت دوره‌ها، و گزارش‌گیری. هر بخش شامل مؤلفه‌ها یا کلاس‌های مرتبط است. توضیحات جزئیتر به شرح زیر است:

1. مدیریت کاربران

این بخش به مدیریت اطلاعات کاربران سیستم اختصاص دارد و شامل دو موجودیت اصلی است:

Student دانشجو

- مسئول نگهداری اطلاعات دانشجویان مانند studentId، name، و سایر جزئیات.

- ممکن است قابلیت‌هایی مانند ثبت‌نام در دوره‌ها یا مشاهده نمرات را داشته باشد.

Teacher استاد

- مسئول مدیریت اطلاعات اساتید مانند teacherId، name، و دوره‌های تدریس‌شده.

- احتمالاً توانایی اختصاص نمرات به دانشجویان یا مدیریت محتوای دوره‌ها را دارد.

2. مدیریت دوره‌ها

این بخش بر سازماندهی دوره‌های آموزشی و نمرات تمرکز دارد و شامل دو مؤلفه است:

- Course دوره شامل اطلاعاتی مانند `courseCode`، `title`، و لیست دانشجویان ثبت‌نام‌شده.
- ممکن است متدهایی برای افزودن/حذف دانشجو یا مدیریت محتوای دوره داشته باشد.
- Grade نمره
- مسئول ثبت و مدیریت نمرات دانشجویان در هر دوره.
- احتمالاً به کلاسهای `Student` و `Course` وابسته است تا نمرات را به دانشجو و دوره مرتبط کند.

3. گزارش‌گیری

این بخش وظیفه تولید گزارشهای مختلف از دادههای سیستم را بر عهده دارد:

- ReportGenerator:
- یک ماژول یا کلاس است که از دادههای بخشهای مدیریت کاربران و مدیریت دورهها استفاده میکند.
- میتواند گزارشهایی مانند لیست دانشجویان یک دوره، کارنامه دانشجو، آمار نمرات، یا خلاصه عملکرد اساتید را تولید کند.
- متدهای احتمالی: `generateStudentReport()`، `generateCourseSummary()`، یا `exportGradesToPDF()`.

ارتباط بین بخشها

1. مدیریت کاربران \Rightarrow مدیریت دوره‌ها:

- دانشجویان (Student) در دوره‌ها (Course) ثبت‌نام میکنند.

- اساتید (Teacher) دوره‌ها را تدریس کرده و نمرات (Grade) را وارد میکنند.

2. گزارشگری \Rightarrow سایر بخشها:

- ReportGenerator از داده‌های Student، Teacher، Course، و Grade برای تولید گزارشها استفاده میکند.

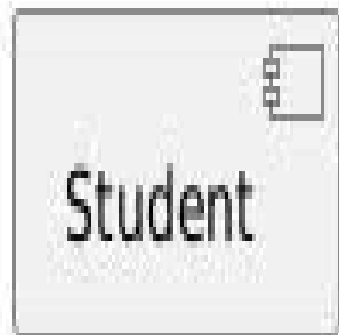
نکات تکمیلی

- عدم نمایش ارتباطات دقیق: در این دیاگرام، جزئیاتی مانند خطوط اتصال، وابستگیها، یا متدهای خاص نشان داده نشده است. این نمودار بیشتر یک نمای سطح بالا از ماژولهای سیستم ارائه میدهد.

- مکانیسمهای پیاده‌سازی: ممکن است کلاس Grade به‌عنوان بخشی از Course یا یک موجودیت مستقل تعریف شود.

- کاربرد عملی: این ساختار برای سیستمهایی مانند سامانه (LMS مدیریت یادگیری)، نرمافزار مدارس، یا پلتفرمهای آموزشی مناسب است.

مدیریت کاربران



مدیریت دوره‌ها



گزارش‌گیری



➤ نمودار مؤلفه (Component Diagram)

- اجزا: مؤلفه‌ها، رابط‌ها، وابستگی‌ها.
- کاربرد: نمایش ساختار فیزیکی سیستم.
- مزایا: وضوح در درک معماری فیزیکی.
- معایب: جزئیات منطقی را نشان نمی‌دهد.
- جنبه‌های ایستایی: ساختار فیزیکی.
- جنبه‌های پویایی: ندارد.

این تصویر یک نمای ساده از معماری یک وب اپلیکیشن را نشان می‌دهد که شامل سه بخش اصلی است. توضیحات هر بخش و ارتباط بین آنها به شرح زیر است:

1. وب اپلیکیشن

این بخش اصلی سیستم است که واسط کاربری (Frontend) و منطق کسبوکار (Backend) را شامل می‌شود. کاربران از طریق مرورگر یا اپلیکیشن موبایل با این بخش تعامل دارند.

2. تأیید کاربر و ذخیره داده‌ها

- تأیید کاربر:
- فرآیند احراز هویت کاربر (Authentication) برای اطمینان از معتبر بودن هویت او.
- مثال: ورود با نام کاربری و رمز عبور، یا استفاده از سرویس‌های احراز هویت خارجی (مانند Google Sign-In).
- ذخیره داده‌ها:
- پس از تأیید کاربر، اطلاعات او (مانند پروفایل، تراکنش‌ها، یا فعالیت‌ها) در پایگاه داده ذخیره می‌شود.
- این بخش ممکن است شامل اعتبارسنجی داده‌ها (Validation) قبل از ذخیره‌سازی باشد.

3. سرور احراز هویت

- یک سرور مجزا که مسئولیت مدیریت فرآیندهای امنیتی را بر عهده دارد:
- ایجاد و بررسی توکنهای امنیتی (مثل JWT).
- مدیریت نشستهای کاربر (Session Management).
- پیادهسازی پروتکلهای امنیتی مانند OAuth 2.0.
- این سرور با وب اپلیکیشن و پایگاه داده در ارتباط است تا اطلاعات کاربر را تأیید و بهروزرسانی کند.

4. دیتابیس (پایگاه داده)

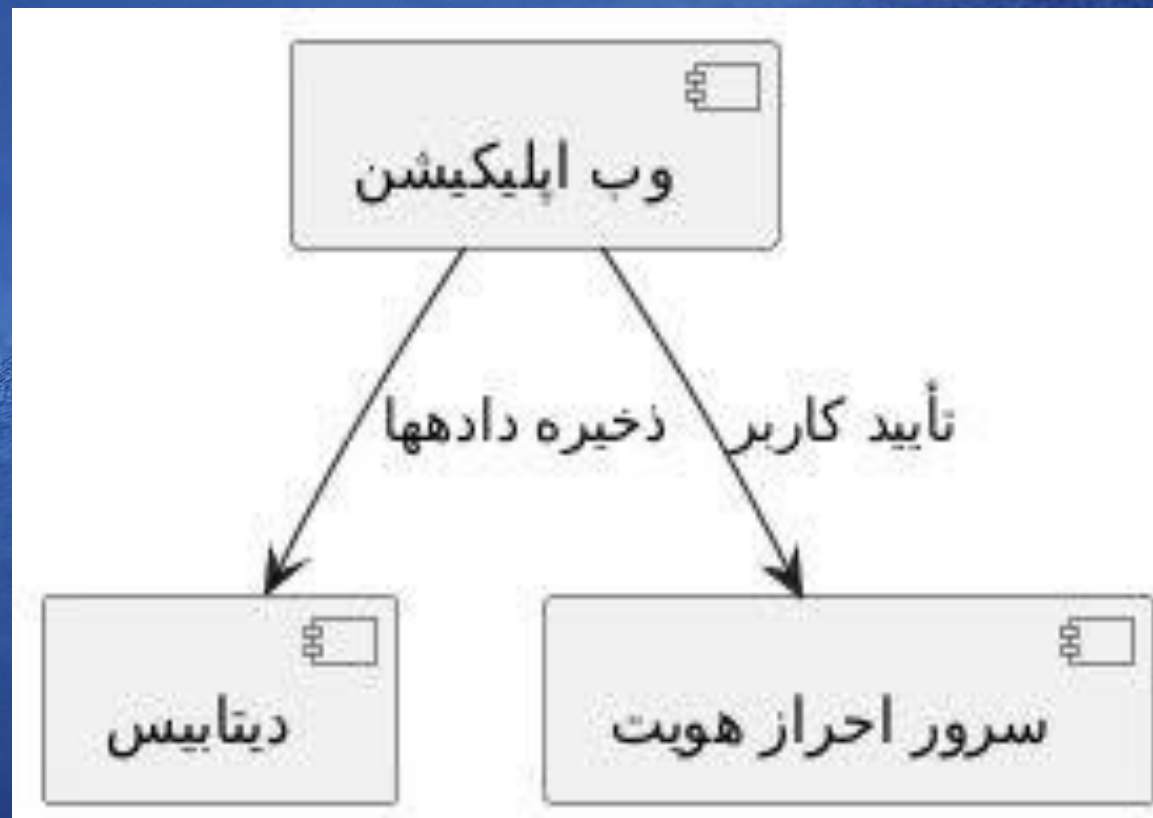
- این بخش مسئول ذخیرهسازی، بازیابی، و مدیریت دادهها است.
- انواع دادههای ذخیرهشده:
- اطلاعات کاربران (پروفایل، رمزهای عبور).
- دادههای مرتبط با عملکرد اپلیکیشن (مانند تراکنشها، محتوای دورهها، نمرات).
- لاگهای سیستم (برای بررسی خطاها یا فعالیتهای مشکوک).
- ممکن است از فناوریهایی مانند MySQL، MongoDB، یا PostgreSQL استفاده شود.

ارتباط بین بخشها

1. وب اپلیکیشن ↔ سرور احراز هویت:
 - کاربر درخواست ورود یا ثبتنام میکند.
 - وب اپلیکیشن درخواست را به سرور احراز هویت ارسال میکند تا هویت کاربر تأیید شود.
2. سرور احراز هویت ↔ دیتابیس:
 - سرور احراز هویت اطلاعات کاربر (مانند رمز عبور یا توکن) را در دیتابیس بررسی یا ذخیره میکند.
3. وب اپلیکیشن ↔ دیتابیس:
 - پس از تأیید هویت، دادههای کاربر (مانند پروفایل یا فعالیتهای آنها) از دیتابیس بازیابی یا در آن ذخیره میشود.

نکات تکمیلی

- امنیت:
- ارتباط بین سرور احراز هویت و دیتابیس باید با پروتکل‌های امن (مثل HTTPS) رمزگذاری شود.
- داده‌های حساس (مانند رمزهای عبور) باید به‌صورت هش‌شده ذخیره شوند.
- مقیاس‌پذیری:
- سرور احراز هویت و دیتابیس می‌توانند به‌صورت جداگانه مقیاس‌پذیر شوند تا بار ترافیکی بالا را مدیریت کنند



➤ نمودار فعالیت (Activity Diagram)

- اجزا: فعالیت‌ها، تصمیم‌گیری‌ها، جریان‌های کنترل.
- کاربرد: نمایش جریان کاری و فرایندها.
- مزایا: وضوح درک فرایندهای کاری.
- معایب: ممکن است به جزئیات زیادی نیاز داشته باشد.
- جنبه‌های ایستایی: ندارد.
- جنبه‌های پویایی: جریان فرایندها.

این تصویر یک فلوجارت فرآیند ثبتنام دانشآموز را نشان میدهد که مراحل زیر را شامل میشود:

1. شروع فرآیند

- دانشآموز فرم ثبتنام را پر میکند (احتمالاً اشتباه تایپی در "داستآموز" و "تتنام" وجود دارد و منظور "دانشآموز فرم ثبتنام را پر میکند" است).

2. بررسی کامل بودن فرم

- سیستم سؤال میکند: آیا فرم کامل است؟
- اگر پاسخ "بله" باشد:
- فرم به مدیر ارسال میشود.
- اگر پاسخ "خیر" باشد:
- فرم به دانشآموز بازگردانده میشود تا آن را اصلاح کند (**دانشآموز فرم را اصلاح میکند**).
- فرآیند به مرحله قبل بازمیگردد (**بازگشت به مرحله قبل**).

3. بررسی فرم توسط مدیر

- مدیر فرم را بررسی میکند.
- سیستم سؤال میکند: "آیا مدیر فرم را تأیید میکند؟"
- اگر پاسخ "بله" باشد:
- اطلاعات در سیستم ذخیره میشود (احتمالاً "دعیره" به معنای "ذخیره" است).
- فرآیند پایان مییابد.
- اگر پاسخ "خیر" باشد:
- پیام خطا به دانشآموز ارسال میشود.
- فرم به دانشآموز بازگردانده میشود تا آن را اصلاح کند.
- فرآیند به مرحله بررسی کامل بودن فرم باز میگردد.

نکات کلیدی و اصلاحات

2. حلقه بازخورد:

- اگر فرم توسط مدیر تأیید نشود یا ناقص باشد، دانشآموز باید آن را اصلاح کند و فرآیند تکرار میشود.

3. ذخیرهسازی دادهها:

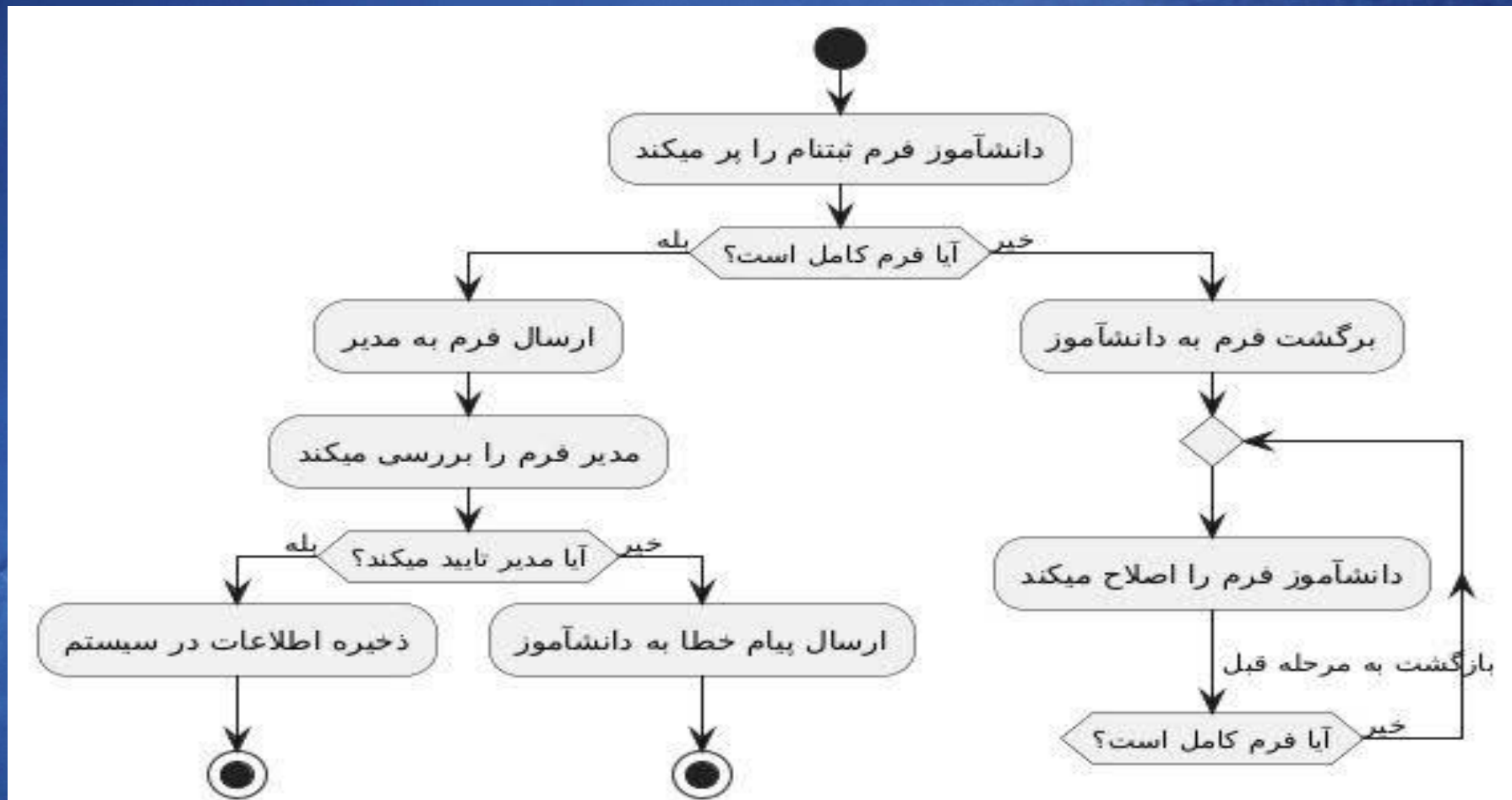
- اطلاعات تنها زمانی ذخیره میشود که مدیر فرم را تأیید کند.

4. عدم نمایش مرحله نهایی:

- پس از تأیید مدیر و ذخیره اطلاعات، فرآیند به صورت خودکار پایان مییابد، اما ممکن است نیاز به افزودن یک گام نهایی (مانند "ثبتنام موفقیت آمیز") برای وضوح بیشتر باشد.

کاربرد فلوچارت

این فلوچارت برای سیستمهای ثبتنام آنلاین مدارس، پلتفرمهای آموزشی، یا سامانههای اداری طراحی شده است. هدف آن استانداردسازی فرآیند ثبتنام و اطمینان از صحت اطلاعات وارد شده توسط دانش آموز است.



➤ نمودار توالی (Sequence Diagram)

- اجزا: اشیاء، پیام‌ها، اکتیویشن‌ها.
- کاربرد: نمایش تعاملات بین اشیاء در طول زمان.
- مزایا: وضوح در درک ترتیب تعاملات.
- معایب: محدودیت در نمایش همزمانی‌ها.
- جنبه‌های ایستایی: ندارد.
- جنبه‌های پویایی: تعاملات و پیام‌ها.

این تصویر یک فرآیند ساده‌ی ثبت‌نام دانشجو در یک سیستم آموزشی را نشان می‌دهد. مراحل اصلی به شرح زیر است:

1. تعامل اولیه:

- دانشجو Student درخواست ثبت نام را با ارسال شناسه دوره Course ID به سیستم ثبت نام ارسال می‌کند.
- سیستم با پایگاه داده (Database**) ارتباط برقرار می‌کند.

2. بررسی دوره:

- سیستم، وجود دوره را در پایگاه داده بررسی می‌کند.
- اگر دوره موجود باشد (**پاسخ (موجود است)**)، اطلاعات دانشجو به دوره اضافه می‌شود افزودن داده یا ثبت اطلاعات دانشجو

3. تأیید نهایی:

- سیستم، ثبتنام را تأیید میکند و فرآیند تکمیل میشود.

نکات مهم:

- این فرآیند، یک جریان ساده برای ثبت نام دانشجو در یک دوره آموزشی با تأیید وجود دوره در پایگاه داده را نشان میدهد

