

نام نویسنده: سجاد غلباش اینالو

عنوان پروژه : طراحی UML

نام درس : مهندسی نرم افزار

نام مدرس : استاد احمدزاده

فهرست:

1.Introduction

2.Use Case

3.Activity

4.Class

5.Object

6.Sequence

زمان و هزینه:

استفاده از پایتون برای توسعه نرم افزار مدیریت دانشگاه می تواند یک انتخاب عالی باشد، زیرا پایتون یک زبان بسیار قابل انعطاف و قدرتمند است. برای تخمین زمان و هزینه با استفاده از پایتون، می توانم مراحل مشابهی را ارائه دهم که قبلاً ذکر شد:

زمان مورد نیاز

1. **تحلیل نیازها** **: ۱-۲ ماه.
2. **طراحی سیستم** **: ۱-۲ ماه.
3. **توسعه و برنامه نویسی با پایتون** **: بسته به پیچیدگی، از ۶ ماه تا ۱ سال.
4. **آزمایش و عیب یابی** **: ۲-۴ ماه.
5. **استقرار و آموزش** **: ۱-۲ ماه.

هزینه ها

1. **هزینه تحلیل و طراحی** **: ۲۰ تا ۵۰ میلیون تومان.
2. **هزینه توسعه و برنامه نویسی با پایتون** **: ۱۰۰ تا ۵۰۰ میلیون تومان.
3. **هزینه آزمایش و عیب یابی** **: ۲۰ تا ۵۰ میلیون تومان.
4. **هزینه استقرار و آموزش** **: ۱۰ تا ۳۰ میلیون تومان.

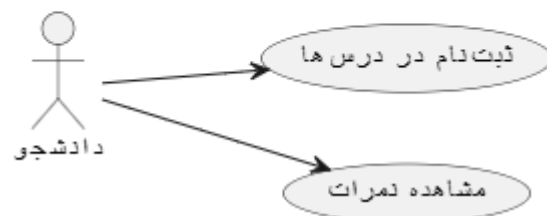
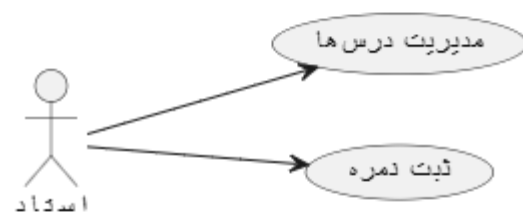
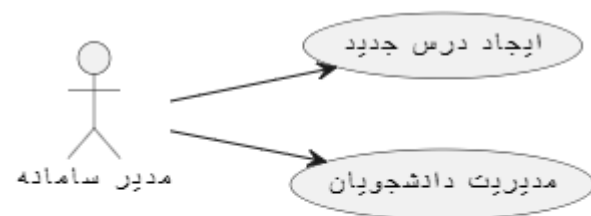
مزایای استفاده از پایتون

1. ****جامعه بزرگ توسعه دهندگان****: پایتون یک جامعه فعال و بزرگ دارد که می‌تواند به حل مشکلات و یافتن منابع کمک کند.
2. ****کتابخانه‌های گسترده****: وجود کتابخانه‌های متعدد مانند Django برای توسعه وب، Pandas برای مدیریت داده‌ها و غیره.
3. ****سازگاری با فناوری‌های مختلف****: پایتون به راحتی با سایر زبان‌ها و فناوری‌ها سازگار است.
4. ****سادگی یادگیری و استفاده****: پایتون زبان نسبتاً ساده‌ای است که یادگیری و استفاده از آن سریع و موثر است.

هزینه‌های جانبی

1. ****خرید سرور و تجهیزات****: نیاز به تجهیزات سخت‌افزاری و سرورهای مناسب.
2. ****هزینه‌های نگهداری و پشتیبانی****: برای حفظ و به‌روزرسانی نرم‌افزار.

تخمین‌ها می‌تواند با توجه به شرایط خاص هر پروژه تغییر کند. برای دقت بیشتر، مشورت با کارشناسان متخصص توصیه می‌شود. آیا برای شروع کار با پایتون سوال دیگری دارید؟



1. ثبت نام در درس‌ها (Registration for Courses)

- توضیح: دانشجو می‌تواند در سامانه در درس‌های ارائه شده برای یک ترم خاص ثبت نام کند.
 - فعالیت‌ها:
 - انتخاب درس از لیست دروس ارائه شده.
 - بررسی پیش نیازها (در صورت نیاز).
 - ثبت موفقیت آمیز درس در پروفایل دانشجو.
 - کاربر مرتبط: دانشجو. (Student)
-

2. مشاهده نمرات (View Grades)

- توضیح: دانشجو می‌تواند نمرات خود را در درس‌های مختلف مشاهده کند.
 - فعالیت‌ها:
 - انتخاب ترم یا درس خاص برای مشاهده نمره.
 - نمایش نمرات به تفکیک درس‌ها.
 - مشاهده اطلاعات اضافی مثل تاریخ امتحان.
 - کاربر مرتبط: دانشجو. (Student)
-

3. مدیریت درس‌ها (Manage Courses)

- توضیح: استاد می‌تواند اطلاعات مربوط به دروس خود را مدیریت کند.
 - فعالیت‌ها:
 - بارگذاری یا به روزرسانی برنامه های کلاسی.
 - اضافه کردن یا حذف منابع مرتبط با درس.
 - ارسال اطلاعیه های مرتبط با درس.
 - کاربر مرتبط: استاد. (Professor)
-

4. ثبت نمره (Record Grades)

- توضیح: استاد می‌تواند نمرات دانشجویان را در سامانه وارد یا به روزرسانی کند.

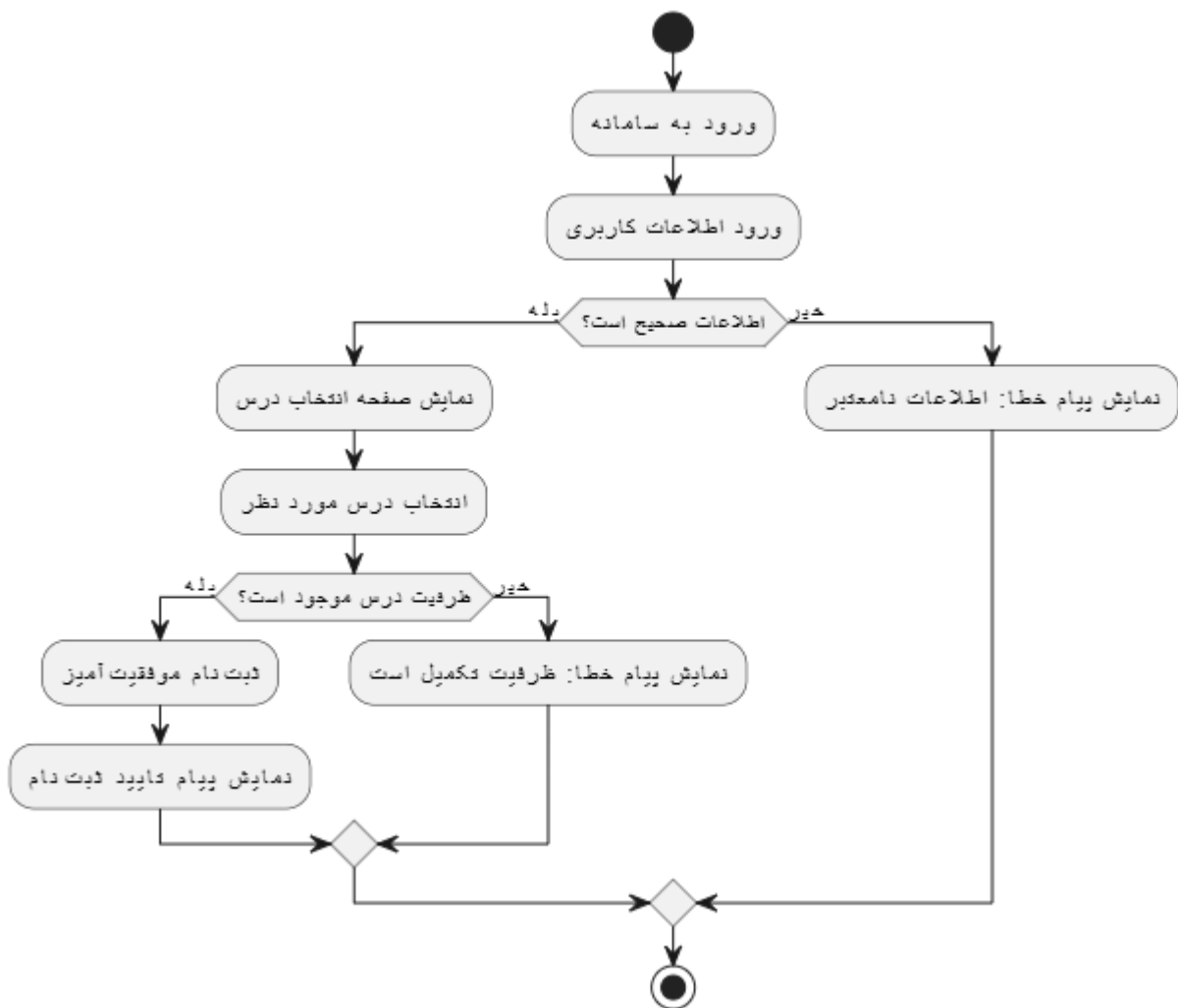
- فعالیت‌ها:
 - وارد کردن نمرات به صورت دستی یا از طریق فایل.
 - تایید و ثبت نمرات برای مشاهده دانشجو.
 - امکان ویرایش نمره در صورت خطا.
 - کاربر مرتبط:
استاد.(Professor)
-

5. ایجاد درس جدید(Create New Course)

- توضیح:
مدیر سامانه می‌تواند درس‌های جدیدی را به سیستم اضافه کند.
 - فعالیت‌ها:
 - تعیین نام درس، کد درس و ظرفیت کلاس.
 - تنظیم پیش‌نیازها یا هم‌نیازهای لازم.
 - تخصیص استاد به درس.
 - کاربر مرتبط:
مدیر سامانه.(Admin)
-

6. مدیریت دانشجویان(Manage Students)

- توضیح:
مدیر سامانه می‌تواند اطلاعات دانشجویان را مدیریت کند.
- فعالیت‌ها:
 - ایجاد یا حذف پروفایل دانشجویان.
 - مدیریت وضعیت تحصیلی (مثلاً تعلیق یا فارغ‌التحصیلی).
 - به‌روزرسانی اطلاعات تماس دانشجویان.
- کاربر مرتبط:
مدیر سامانه.(Admin)



· شروع فرآیند: (start) فرآیند با ورود به سامانه آغاز می‌شود.

· ورود اطلاعات کاربری:

کاربر اطلاعات کاربری (نام کاربری و رمز عبور) خود را وارد می‌کند.

· بررسی اطلاعات کاربری (اطلاعات صحیح است?):

سیستم بررسی می‌کند که آیا اطلاعات وارد شده معتبر است یا خیر.

· اگر اطلاعات صحیح باشد:

○ کاربر به صفحه انتخاب درس هدایت می‌شود.

· اگر اطلاعات اشتباه باشد:

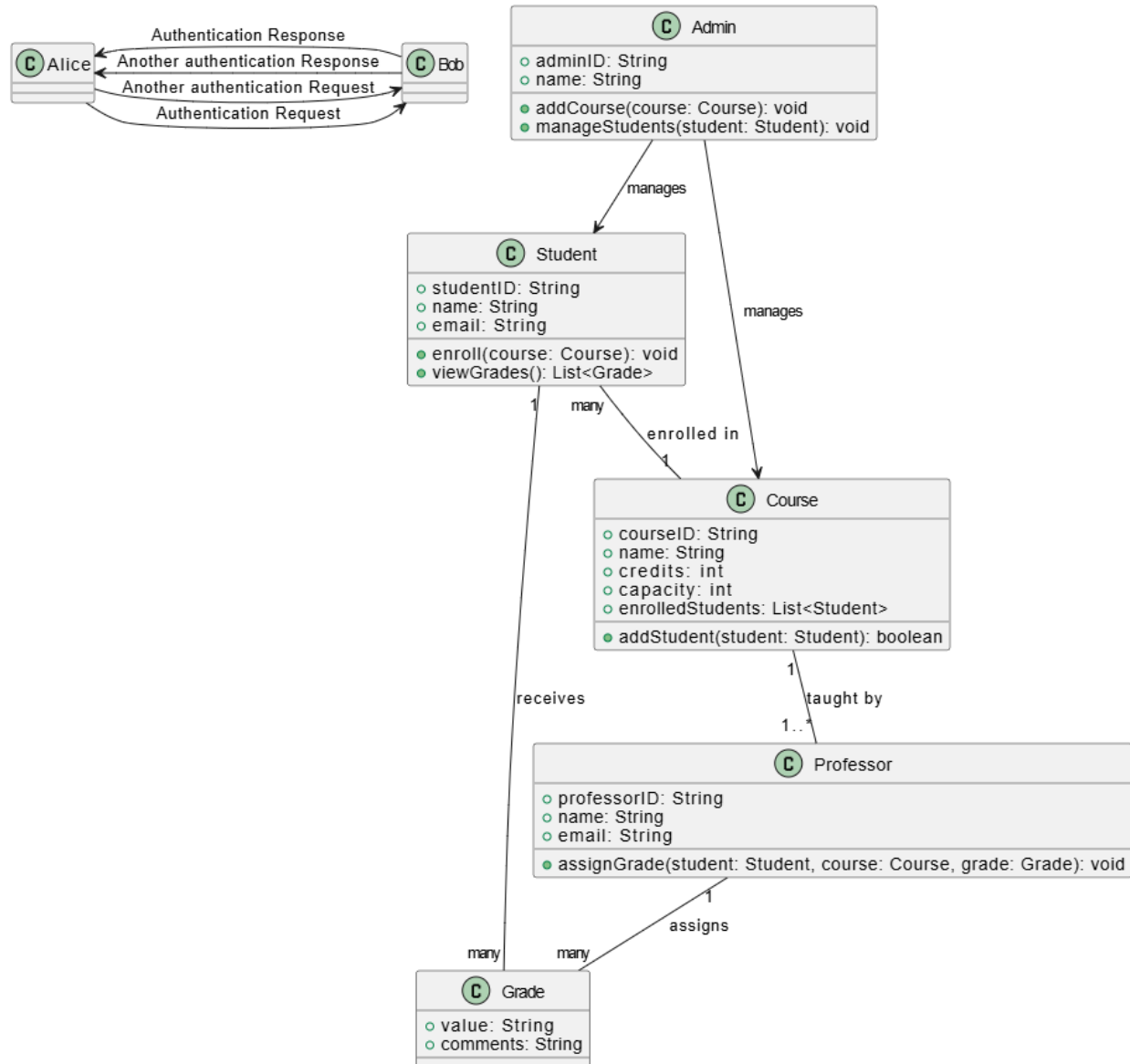
○ پیامی مبنی بر نامعتبر بودن اطلاعات نمایش داده می‌شود و فرآیند متوقف می‌شود.

• انتخاب درس:
کاربر درس مورد نظر خود را از لیست دروس انتخاب می کند.

• بررسی ظرفیت درس (ظرفیت درس موجود است؟):
سیستم بررسی می کند که آیا ظرفیت درس انتخاب شده پر شده است یا خیر.

- اگر ظرفیت موجود باشد:
 - ثبت نام انجام می شود و پیام تایید نمایش داده می شود.
- اگر ظرفیت تکمیل شده باشد:
 - پیام خطا به کاربر نمایش داده می شود.

• پایان فرآیند: (stop)
فرآیند ثبت نام خاتمه می یابد.



توضیحات کد:

1. کلاسها:

Student دانشجو: (

▪ مشخصات:

▪ studentID: شماره دانشجویی.

▪ name: نام دانشجو.

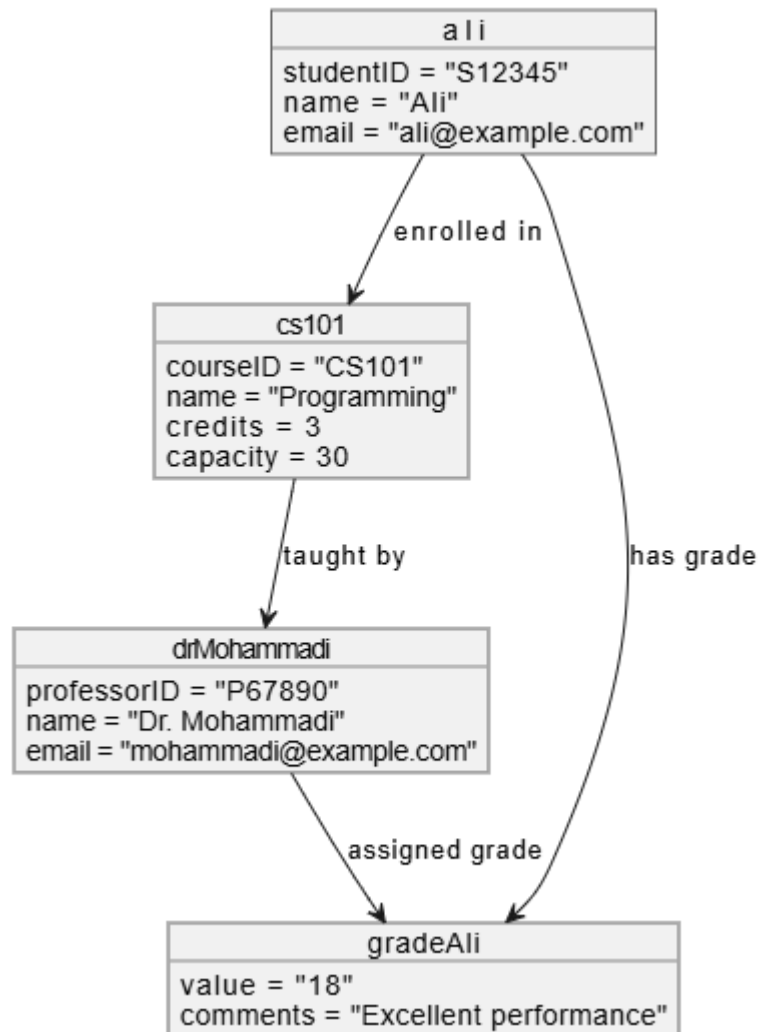
▪ email: ایمیل دانشجو.

▪ عملیات:

- enroll(course: Course): ثبت نام در درس.
- viewGrades(): مشاهده نمرات.
- Professor (استاد):
 - مشخصات:
 - professorID: شناسه استاد.
 - name: نام استاد.
 - email: ایمیل استاد.
 - عملیات:
- assignGrade(student: Student, course: Course, grade: Grade): اختصاص نمره به دانشجو.
- Course (درس):
 - مشخصات:
 - courseID: کد درس.
 - name: نام درس.
 - credits: تعداد واحد درس.
 - capacity: ظرفیت درس.
 - enrolledStudents: لیست دانشجویان ثبت نام شده.
 - عملیات:
- addStudent(student: Student): افزودن دانشجو به درس.
- Grade (نمره):
 - مشخصات:
 - value: مقدار نمره (مثل "A" یا "20").
 - comments: توضیحات استاد.
 - Admin (مدیر):
 - مشخصات:
 - adminID: شناسه مدیر.
 - name: نام مدیر.
 - عملیات:
 - addCourse(course: Course): افزودن درس جدید.
 - manageStudents(student: Student): مدیریت دانشجویان.

2. روابط: (Relationships)

- دانشجو (Student) با کلاس درس (Course) در ارتباط است:
 - یک دانشجو می تواند در چند درس ثبت نام کند، و یک درس می تواند چند دانشجو داشته باشد (ارتباط چند به یک).
- استاد (Professor) با کلاس درس (Course):
 - یک استاد می تواند یک یا چند درس را تدریس کند.
- نمره (Grade):
 - یک دانشجو می تواند چند نمره داشته باشد.
 - یک استاد نمرات دانشجویان را ثبت می کند.
- مدیر (Admin):
 - مدیر سامانه وظیفه مدیریت درس ها و دانشجویان را دارد.



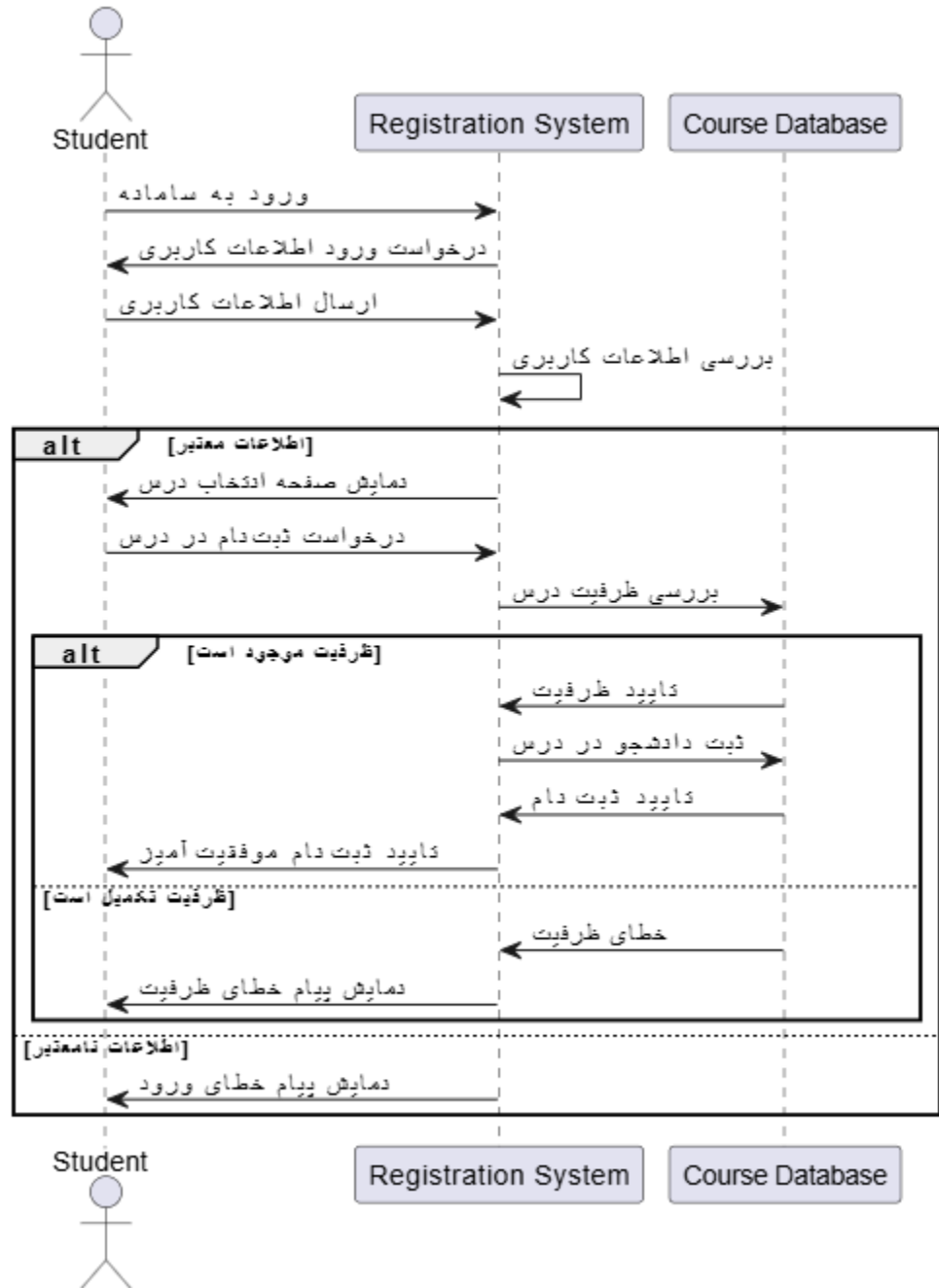
1. شیء ali از نوع Student:
 - نمایانگر یک دانشجوی خاص است.
 - ویژگی‌های دانشجو:
 - شماره دانشجویی (studentID): S12345
 - نام (name): Ali
 - ایمیل (email): ali@example.com
 - این شیء مشخص می‌کند که این دانشجو ثبت‌نام کرده است.
 2. شیء cs101 از نوع Course:
 - نماینده یک درس خاص در دانشگاه است.
 - ویژگی‌های درس:
 - شناسه درس (courseID): CS101
 - نام (name): Programming (نام برنامه‌نویسی)
 - تعداد واحد (credits): 3
 - ظرفیت درس (capacity): 30
 - این درس مشخصاً توسط یک استاد خاص تدریس می‌شود.
 3. شیء drMohammadi از نوع Professor:
 - نماینده استادی است که درس cs101 را تدریس می‌کند.
 - ویژگی‌های استاد:
 - شماره استاد (professorID): P67890
 - نام (name): Dr. Mohammadi (نام دکتر محمدی)
 - ایمیل (email): mohammadi@example.com
 4. شیء gradeAli از نوع Grade:
 - این شیء نشان‌دهنده نمره‌ای است که توسط استاد drMohammadi به دانشجوی ali برای درس cs101 داده شده است.
 - ویژگی‌های نمره:
 - مقدار نمره (value): 18
 - توضیحات (comments): Excellent performance (عملکرد عالی)
-

روابط بین اشیاء:

1. ali --> cs101 : "enrolled in":
 - این رابطه نشان می‌دهد که دانشجوی ali در درس cs101 ثبت‌نام کرده است.
2. cs101 --> drMohammadi : "taught by":
 - این رابطه نشان می‌دهد که درس cs101 توسط استاد drMohammadi تدریس می‌شود.
3. ali --> gradeAli : "has grade":
 - این رابطه نشان می‌دهد که دانشجوی ali نمره‌ای مرتبط با درس دریافت کرده است.
4. drMohammadi --> gradeAli : "assigned grade":
 - این رابطه نشان می‌دهد که استاد drMohammadi نمره را برای دانشجوی ali تعیین کرده است.

کاربرد این نمودار:

- تحلیل وضعیت لحظه‌ای:
وضعیت یک دانشجو، یک درس، و نمرات مربوطه در لحظه مشخص.
- نمایش تعامل بین اجزاء سیستم:
نحوه ارتباط دانشجو، استاد، و درس.
- مدل‌سازی سیستم آموزشی:
استفاده از این نمودار برای فهم روابط در سیستم ثبت‌نام دانشگاهی.



1. بازیگران و اجزاء:
 - **Student** (دانشجو): کاربر سامانه که درخواست ثبت نام می دهد.
 - **Registration System** (سیستم ثبت نام): سامانه ای که عملیات ثبت نام را مدیریت می کند.

- بخشی از سیستم که اطلاعات درس‌ها، ظرفیت‌ها و (پایگاه داده درس) **Course Database** ثبت‌نام‌ها را مدیریت می‌کند.
- 2. **جریان عملیات:**
 - **ورود دانشجو:**
 - دانشجو به سامانه وارد می‌شود و اطلاعات کاربری خود را ارسال می‌کند.
 - سیستم اطلاعات کاربری را بررسی می‌کند.
 - در صورت صحت اطلاعات، صفحه انتخاب درس نمایش داده می‌شود.
 - در غیر این صورت، پیام خطای ورود نمایش داده می‌شود.
 - **ثبت‌نام درس:**
 - دانشجو درخواست ثبت‌نام در یک درس خاص را ارسال می‌کند.
 - سیستم ثبت‌نام ظرفیت درس را از پایگاه داده بررسی می‌کند.
 - اگر ظرفیت موجود باشد:
 - دانشجو در درس ثبت می‌شود و پیام تایید ارسال می‌شود.
 - اگر ظرفیت تکمیل باشد:
 - پیام خطای ظرفیت نمایش داده می‌شود.
- 3. **ساختارهای شرطی:**
 - **alt:**

برای نمایش شرط‌های مختلف استفاده شده است

 - در صورت معتبر بودن اطلاعات کاربری
 - در صورت وجود یا عدم وجود ظرفیت
- 4. **پیام‌ها:**
 - نشان داده شده‌اند (>-) با پیکان: **پیام‌های مستقیم**
 - پیام‌هایی که سیستم به دانشجو یا پایگاه داده باز می‌گرداند: **پاسخ‌ها**