

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

مهندسی نرم افزار ۱

"فاز 6 پروژه"

موضوع پروژه: Wiki Service

استاد درس: دکتر کلباسی

نام گروه:

Nullterminated

1_ توضیحات جامع نمودارهای UML پروژه Wiki Service

1.1 منابع اصلی استخراج

نمودارهای طراحی شده از سه فاز قبلی پروژه استخراج شده‌اند:

فاز 3: داستان‌های کاربر (User Stories) :

- US-01 : جستجوی مقاله تبدیل به " Use Case جستجوی مقاله "
- US-02 : مشاهده مقاله با پیوندهای مرتبط تبدیل به " Use Case مشاهده مقاله "
- US-03 : ایجاد مقاله جدید تبدیل به " Use Case ایجاد مقاله جدید "
- US-04 : ویرایش مقاله تبدیل به " Use Case ویرایش مقاله "
- US-05 : مشاهده تاریخچه تبدیل به " Use Case مشاهده تاریخچه "
- US-07 : تولید برچسب خودکار تبدیل به " Use Case سیستمی "
- US-08 : گزارش مقاله نامناسب تبدیل به " Use Case گزارش مقاله "
- US-09 تبدیل به فیلتر مقالات " Use Case فیلتر نتایج "
- US-10 : اعلان تغییرات تبدیل به " Use Case دریافت اعلان "

فاز 3: نیازمندی‌های عملکردی (Functional Requirements)

از بخش نیازمندی‌ها:

- FR-1.1 تا FR-1.3 : جستجوی معنایی و فیلتر تبدیل به کلاس‌های SearchService و SemanticSearchEngine
- FR-2.1 تا FR-2.2 : نمایش مقاله و پیوندها تبدیل به کلاس‌های ArticleService و ArticleLink
- FR-3.1 تا FR-3.2 : ایجاد مقاله تبدیل به متدهای createArticle در ArticleService
- FR-4.1 تا FR-4.2 : ویرایش و مقایسه نسخه‌ها تبدیل به کلاس RevisionService
- FR-7.1 : تولید برچسب خودکار تبدیل به کلاس TagService و NLPProcessor
- FR-8.1 : گزارش مقاله تبدیل به کلاس ReportService
- FR-12.1 و FR-13.1 : خلاصه‌سازی و دسته‌بندی با LLM تبدیل به کلاس LLMService

فاز 5: طراحی پایگاه داده

• جداول دیتابیس مستقیماً به کلاس‌های Domain Model تبدیل شدند :

- wiki_articles تبدیل به کلاس Article
- wiki_categories تبدیل به کلاس Category
- wiki_tags تبدیل به کلاس Tag
- wiki_article_revisions تبدیل به کلاس Revision
- wiki_article_ratings تبدیل به کلاس Rating
- wiki_article_reports تبدیل به کلاس Report
- wiki_article_links تبدیل به کلاس ArticleLink

فاز 5: معماری سیستم

معماری Layered Architecture انتخاب شده به سه لایه تقسیم شد:

1. Presentation Layer → Controllers (ArticleController, SearchController, ...)
2. Business Logic Layer → Services (ArticleService, SearchService, ...)
3. Data Access Layer → Repositories (ArticleRepository, CategoryRepository, ...)

2_ نقشه‌برداری مستقیم User Story به Use case

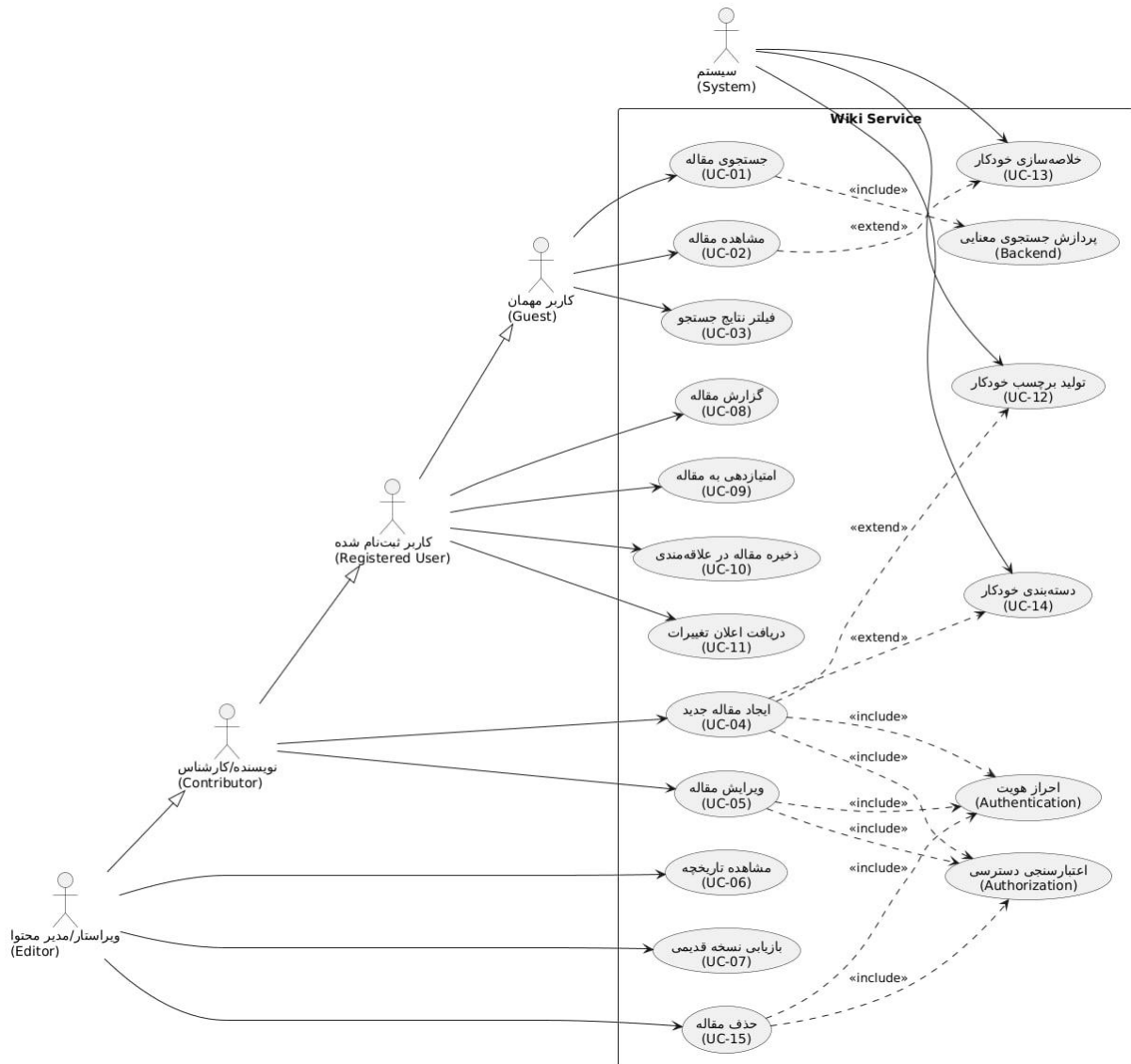
User Story	شماره UC	نام Use Case	بازیگر	نیازمندی مرتبط
US-01	UC-01	جستجوی مقاله	Guest/User	FR-1.1, FR-1.2, FR-1.3
US-02	UC-02	مشاهده مقاله	Guest/User	FR-2.1, FR-2.2
US-03	UC-04	ایجاد مقاله جدید	Contributor	FR-3.1, FR-3.2
US-04	UC-05	ویرایش مقاله	Contributor/Editor	FR-4.1
US-05	UC-06	مشاهده تاریخچه	Editor	FR-4.2, FR-5.1
US-07	UC-12	تولید برچسب خودکار	System	FR-7.1
US-08	UC-08	گزارش مقاله	User	FR-8.1
US-09	UC-03	فیلتر مقالات	User	FR-1.3, FR-10.1
US-10	UC-11	دریافت اعلان	User	FR-9.1

3_ نقشه‌برداری دیتابیس به Domain Model

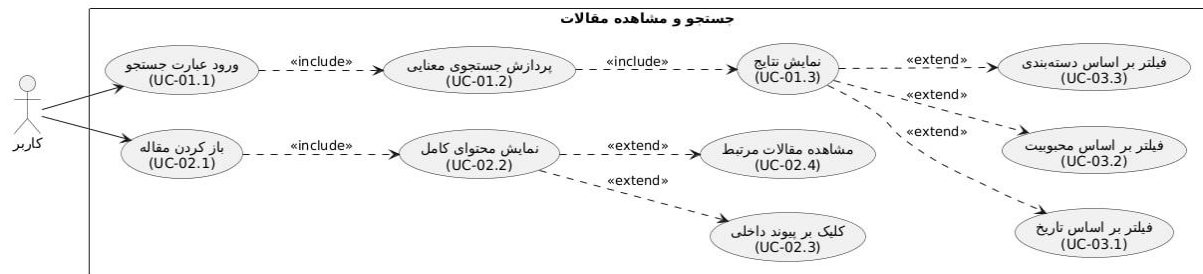
روابط	Attributes کلیدی	کلاس Domain	جدول دیتابیس
1-* Revision, - Tag, *-1 Category	id, titleFa, bodyFa, status, qualityAvg	Article	wiki_articles
Self-reference (سلسله‌مراتبی)	id, titleFa, parentId	Category	wiki_categories
- Article	id, slug, titleFa	Tag	wiki_tags
*-1 Article	id, articleId, revisionNo, bodyFa	Revision	wiki_article_revisions
*-1 Article	articleId, userId, rating (ترکیبی) PK	Rating	wiki_article_ratings
*-1 Article	id, articleId, reporterUserId, status	Report	wiki_article_reports
*-1 Article (دوطرفه)	fromArticleId, toArticleId	ArticleLink	wiki_article_links

4_نمودار های USE CAS :

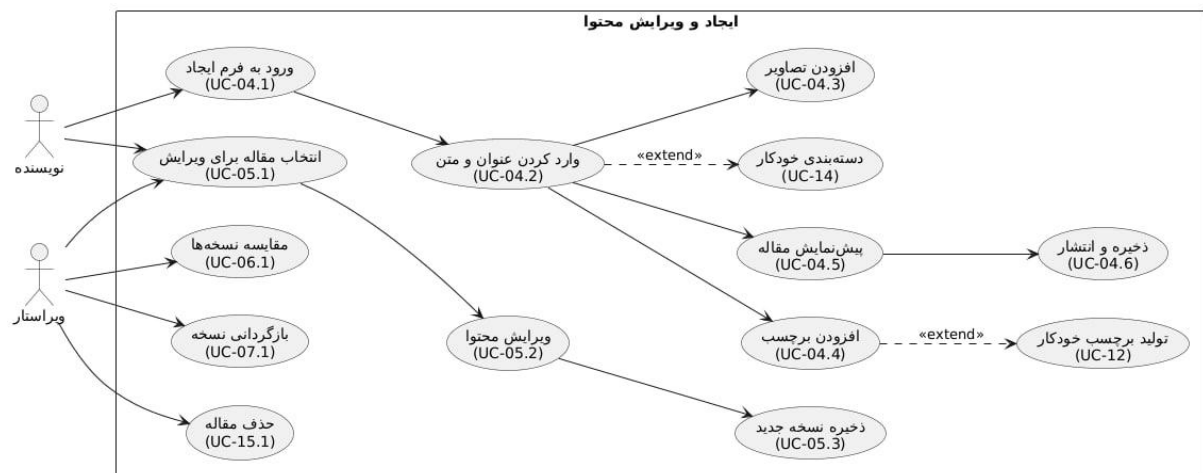
- نمودار USE CASE کلی :



• نمودار USE CASE مربوط به جستجو و مشاهده مقالات:



• نمودار USE CASE مربوط به ایجاد و ویرایش محتوا:



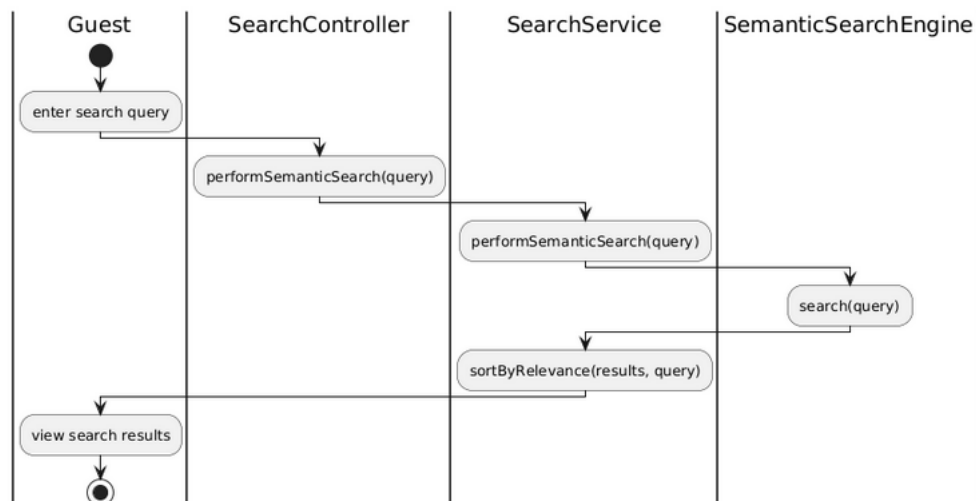
5_نمودار : Class Diagram

این نمودار به دلیل ابعاد بزرگ، در فایل جداگانه ارائه شده است. برای مشاهده نمودار کلاس، به فایل `uml_class_diagram` در فایل زیپ مراجعه کنید.

6_نمودار های Activity Diagram :

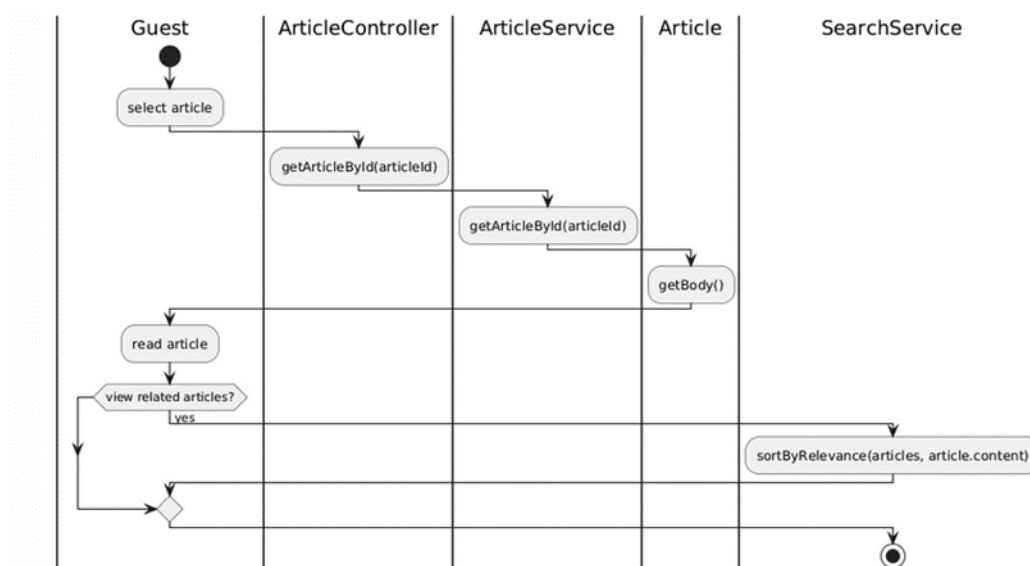
• US-01 — Search Article

در این سناریو، کاربر با وارد کردن عبارت جستجو درخواست خود را ثبت می کند. سیستم از طریق `SearchController` و `SearchService` جستجوی معنایی را انجام داده و نتایج مرتبط را مرتب سازی کرده و به کاربر نمایش می دهد. این فرآیند با استفاده از موتور جستجوی معنایی انجام شده و با معماری لایه ای سیستم سازگار است.



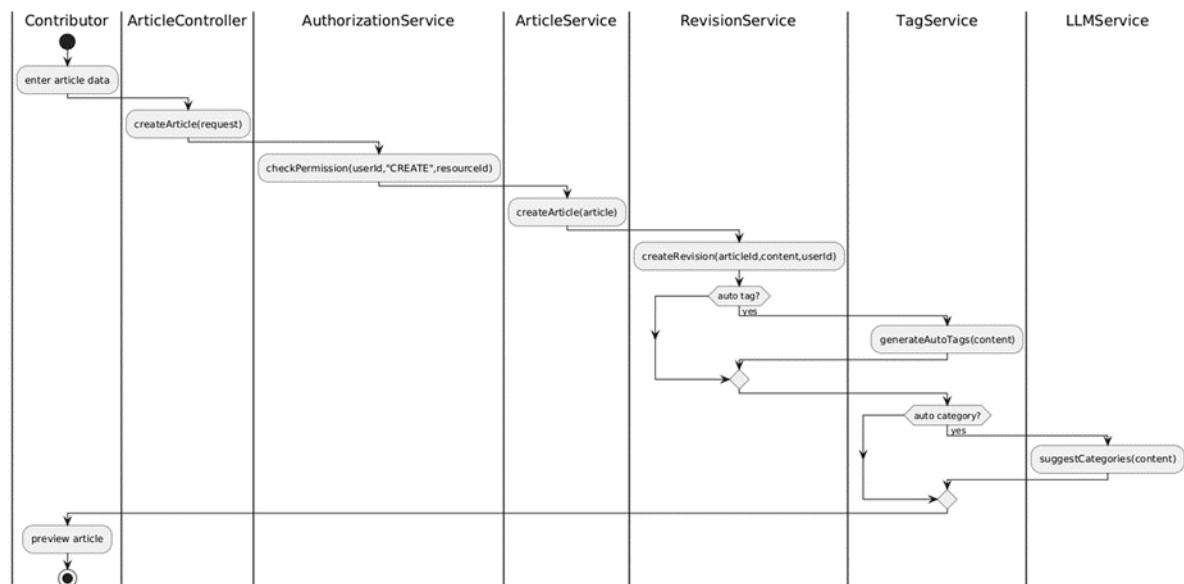
US-02 — View Article •

کاربر با انتخاب یک مقاله، محتوای کامل آن را مشاهده می‌کند. درخواست از طریق `ArticleController` به `ArticleService` ارسال شده و اطلاعات مقاله بازیابی می‌شود. در صورت تمایل، کاربر می‌تواند مقالات مرتبط یا پیوندهای داخلی را نیز مشاهده کند. این سناریو با `Use Case` های توسعه یافته مرتبط با مشاهده مقاله هم‌راستا است.



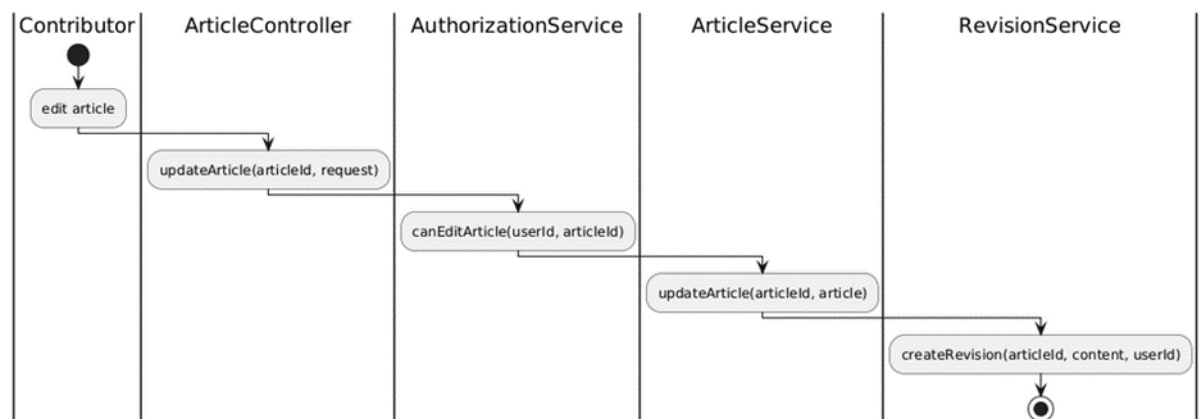
US-03 — Create New Article •

در این سناریو، نویسنده اقدام به ایجاد مقاله جدید می‌کند. پس از بررسی دسترسی کاربر توسط `AuthorizationService`، مقاله از طریق `ArticleService` ایجاد شده و یک نسخه اولیه توسط `RevisionService` ذخیره می‌شود. در ادامه، سیستم می‌تواند به صورت خودکار برچسب‌ها و دسته‌بندی مقاله را تولید کند.



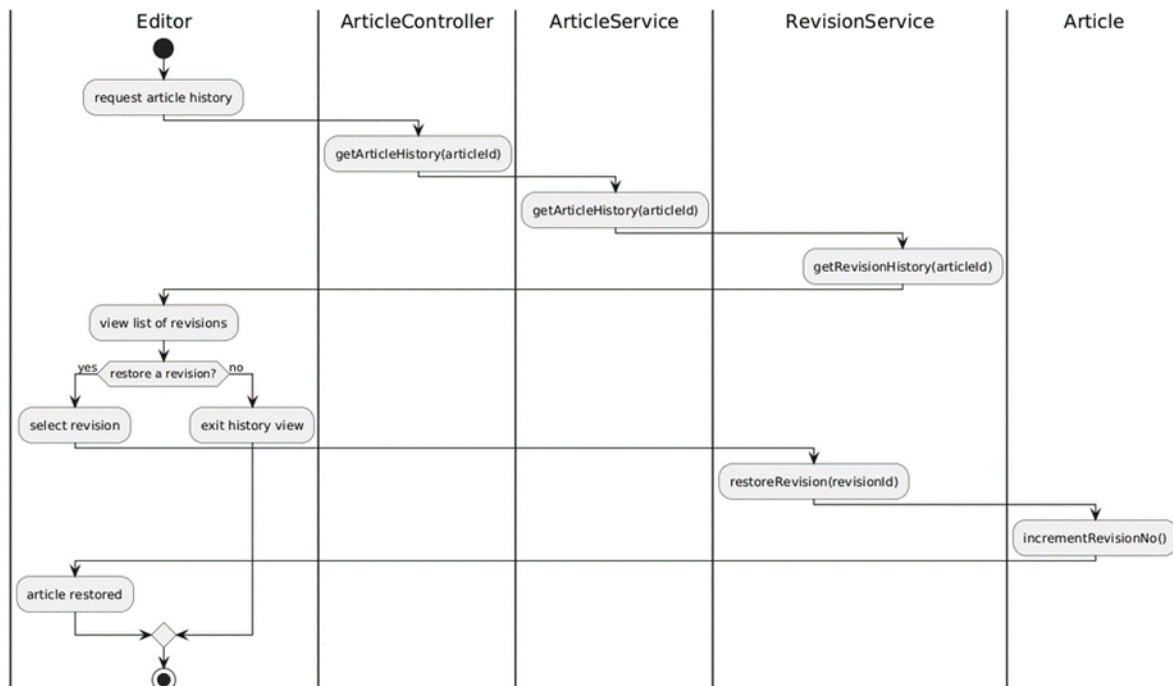
US-04 — Edit Article •

نویسنده یا ویراستار مقاله‌ای موجود را ویرایش می‌کند. ابتدا مجوز ویرایش بررسی شده و سپس محتوای جدید ذخیره می‌شود. پس از ویرایش، یک نسخه جدید از مقاله در سیستم ثبت می‌گردد تا تاریخچه تغییرات حفظ شود.



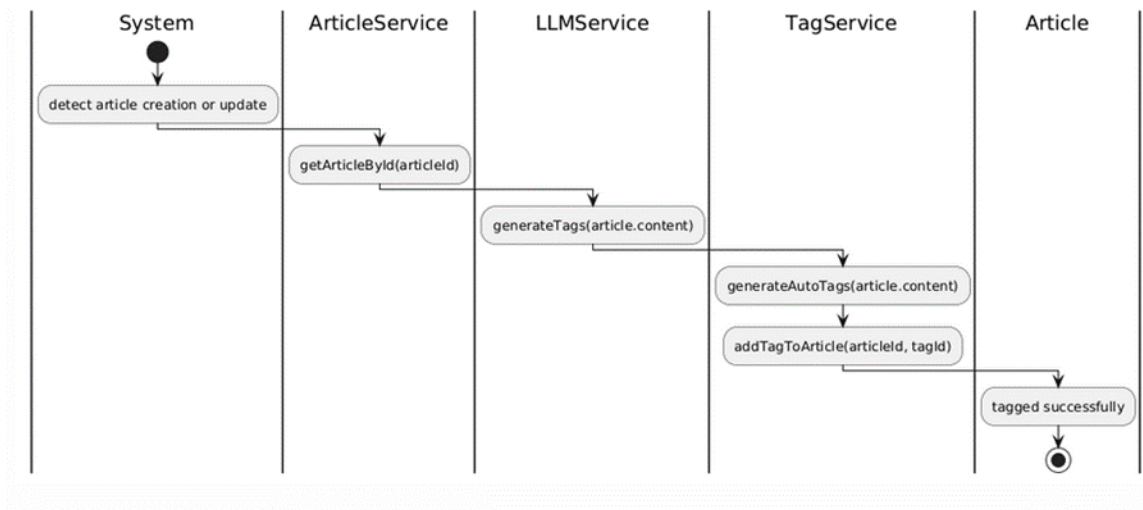
US-05 — View History / Compare Versions •

در این سناریو، ویراستار تاریخچه نسخه‌های یک مقاله را مشاهده می‌کند. در صورت نیاز، می‌تواند یکی از نسخه‌های قبلی را انتخاب و بازیابی کند. بازیابی نسخه به‌عنوان یک مسیر اختیاری پس از مشاهده تاریخچه انجام می‌شود و باعث بازگشت مقاله به نسخه انتخاب‌شده می‌گردد.



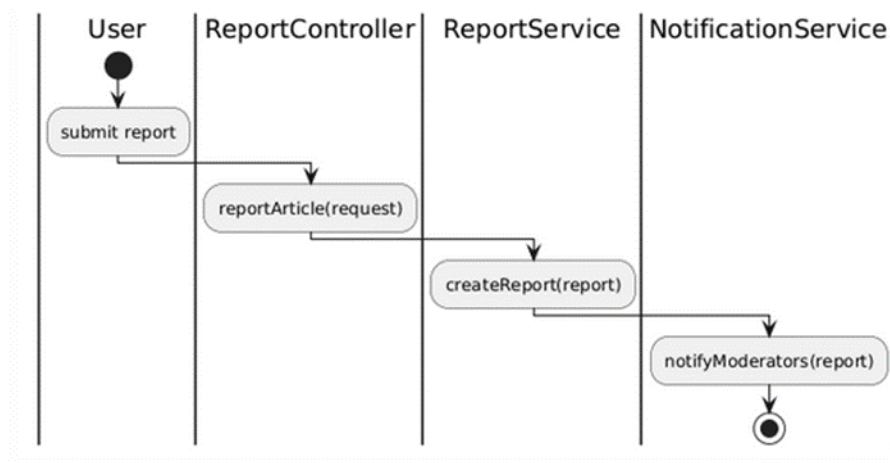
US-07 — Auto-generate Tags •

در این سناریو، سیستم پس از ایجاد یا ویرایش مقاله، محتوای آن را تحلیل کرده و برچسب‌های مناسب را به‌صورت خودکار تولید می‌کند. این فرآیند با استفاده از LLMService انجام شده و سپس برچسب‌ها از طریق TagService به مقاله متصل می‌شوند.



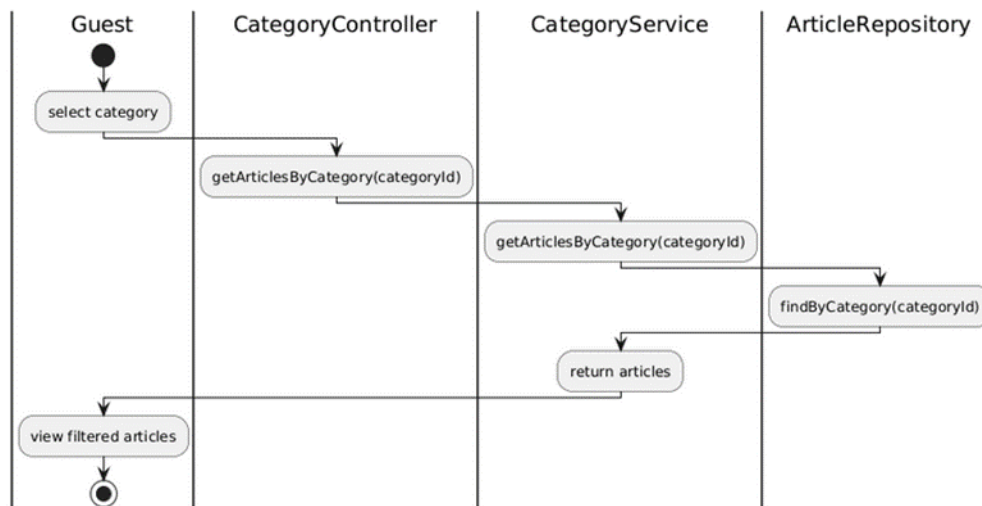
US-08 — Report Article •

کاربر ثبت نام شده می تواند مقاله ای را گزارش کند. گزارش از طریق ReportController ثبت شده و توسط ReportService ذخیره می شود. پس از ثبت گزارش، اعلان لازم برای مدیران محتوا ارسال می گردد تا بررسی انجام شود.



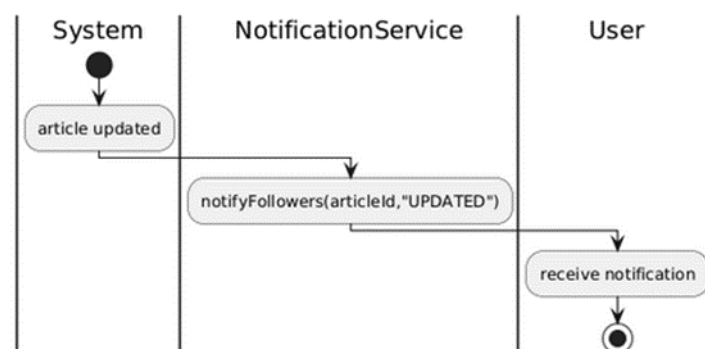
US-09 — Filter Articles •

در این سناریو، کاربر یک دسته‌بندی را انتخاب می‌کند تا فقط مقالات مرتبط با آن نمایش داده شوند. CategoryController درخواست را دریافت کرده و از طریق CategoryService مقالات مربوطه بازیابی می‌شوند. نتایج فیلترشده در نهایت به کاربر نمایش داده می‌شوند.



• US-10_Receive Article Change Notifications

در این سناریو، پس از ایجاد یا ویرایش مقاله، سیستم به‌صورت خودکار تغییرات را شناسایی کرده و از طریق NotificationService به کاربران دنبال‌کننده اعلان ارسال می‌کند. کاربر اعلان را دریافت کرده و می‌تواند به مقاله دسترسی پیدا کند.



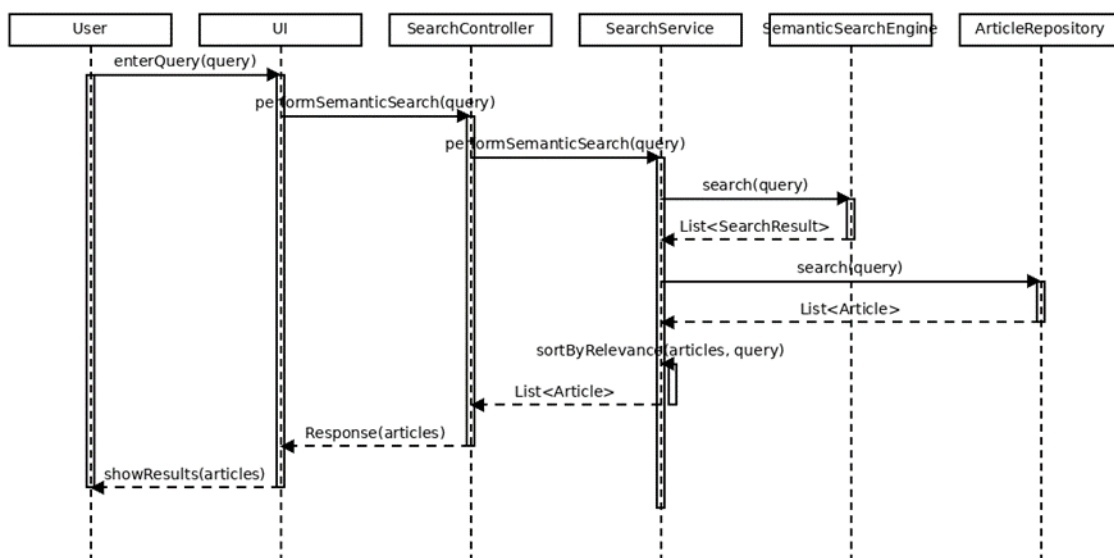
7_ نمودار های Sequence Diagram :

• US-01 — Search Article (Semantic Search)

جستجوی مقاله به صورت معنایی از مسیر زیر انجام میشود و سپس نتایج مرتب سازی میشوند

UI → SearchController → SearchService → SemanticSearchEngine → ArticleRepository

US-01 — Search Article (Semantic Search)

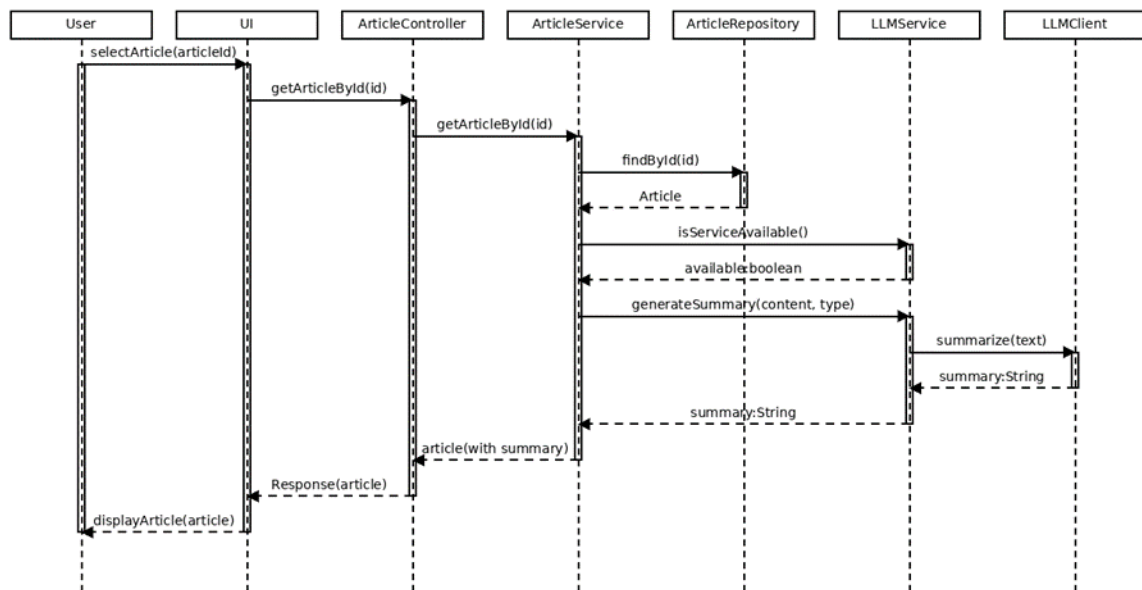


• US-02 — View Article (with AI Summary)

مشاهده مقاله با خواندن از ArticleRepository انجام می شود و در صورت فعال بودن سرویس هوش

مصنوعی، خلاصه از طریق LLMService/LLMClient تولید می گرد

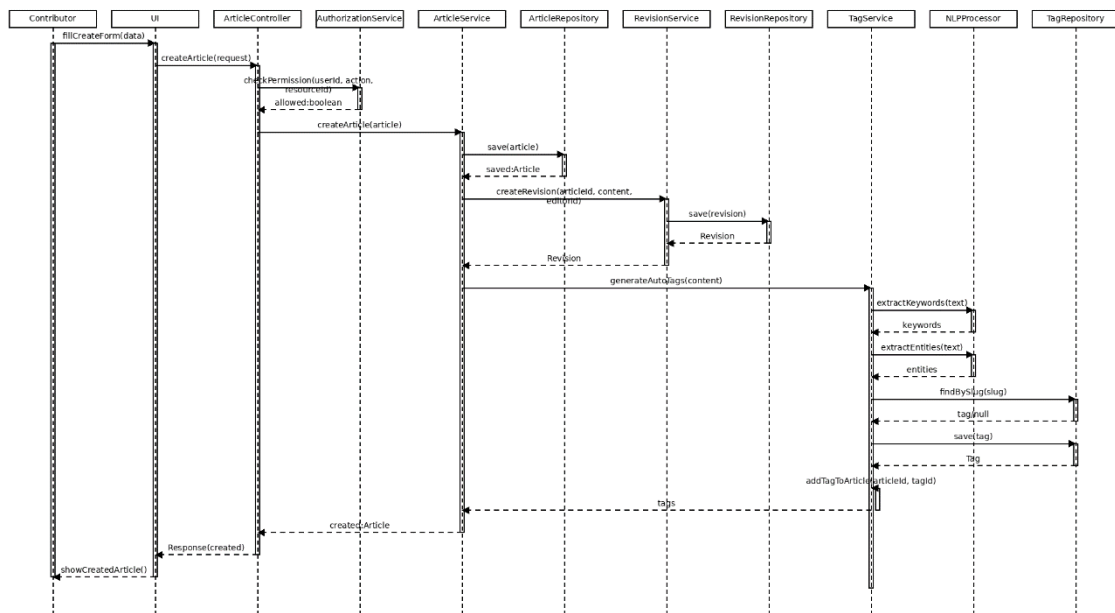
US-02 — View Article (with AI Summary)



US-03 — Create New Article •

ایجاد مقاله جدید شامل کنترل دسترسی (AuthorizationService)، ذخیره مقاله (ArticleRepository)، ثبت نسخه اولیه (RevisionService/RevisionRepository) و تولید برچسب خودکار (TagService/NLPProcessor/TagRepository) است

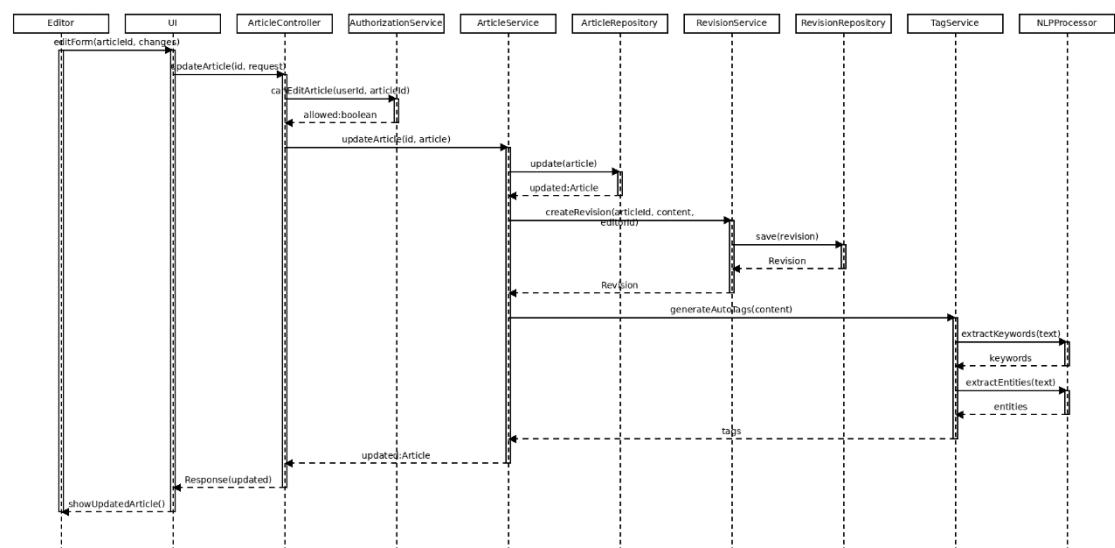
US-03 — Create New Article



US-04 — Edit Article •

ویرایش مقاله شامل بررسی مجوز، به‌روزرسانی مقاله، ایجاد Revision جدید و (اختیاری) بازتولید برچسب‌های خودکار است.

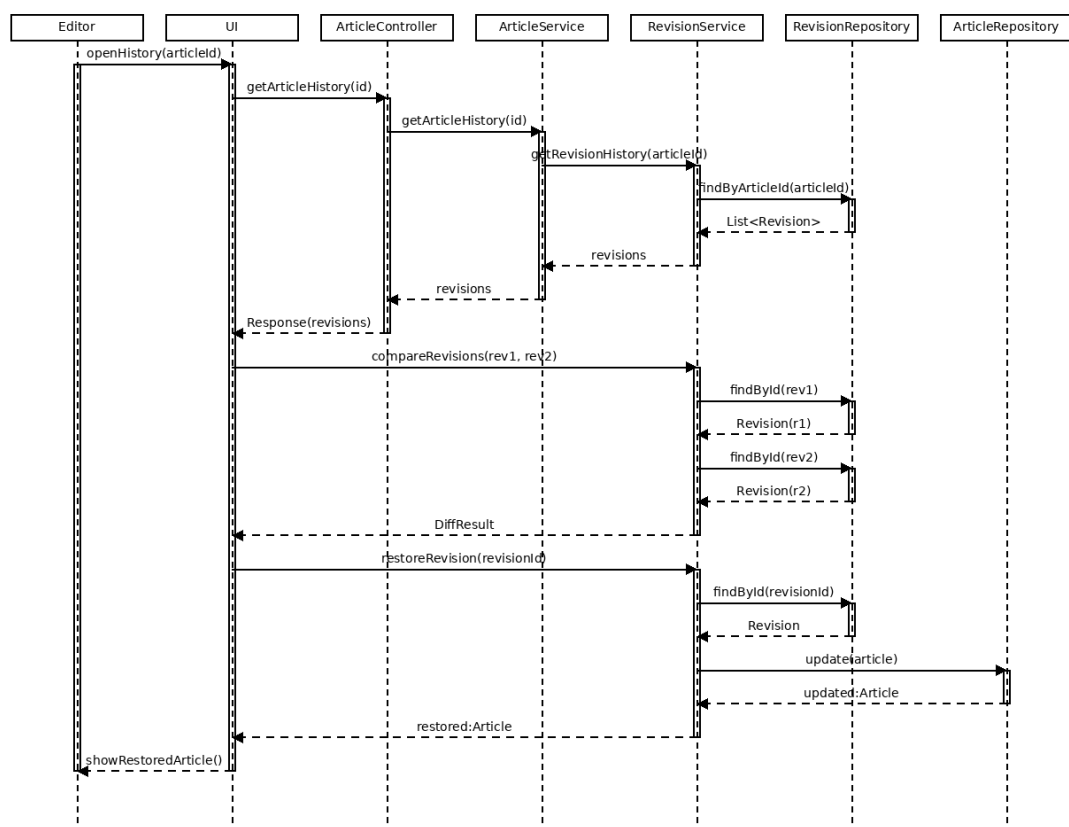
US-04 — Edit Article



US-05 — View History / Compare / Restore •

تاریخچه نسخه‌ها از RevisionRepository خوانده می‌شود. سپس امکان مقایسه دو نسخه و بازیابی نسخه انتخاب‌شده نمایش داده شده است.

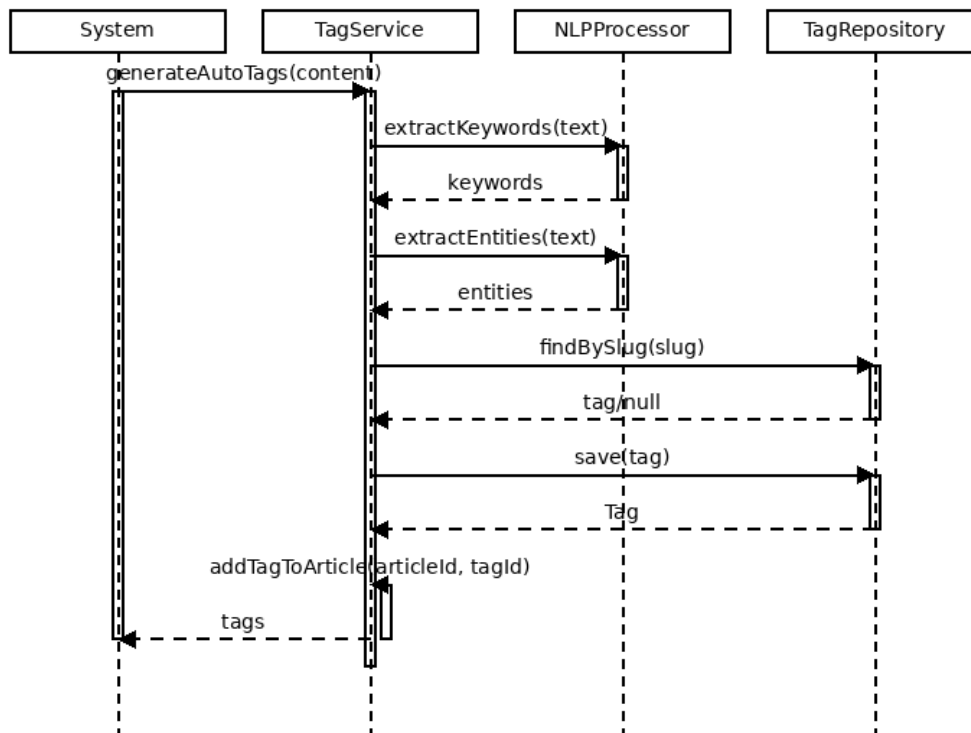
US-05 — View History / Compare / Restore



US-07 — Auto-generate Tags •

تولید برچسب خودکار توسط TagService با کمک NLPPProcessor انجام می‌شود و سپس TagRepository برای یافتن/ذخیره برچسب‌ها استفاده می‌شود.

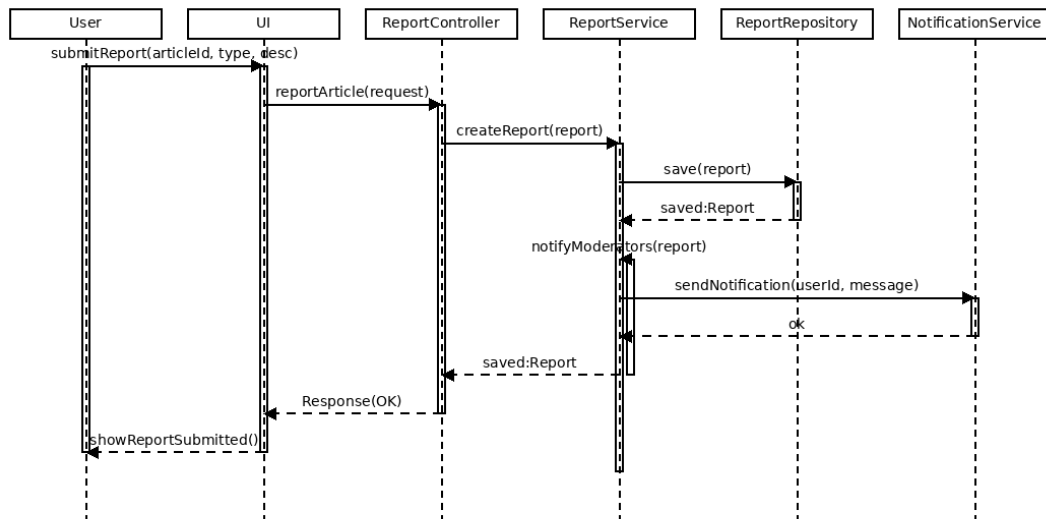
US-07 — Auto-generate Tags



US-08 — Report Article •

ثبت گزارش توسط کاربر از مسیر `ReportController/ReportService` انجام می‌شود و برای اطلاع مدیران، `NotificationService` فراخوانی می‌گردد.

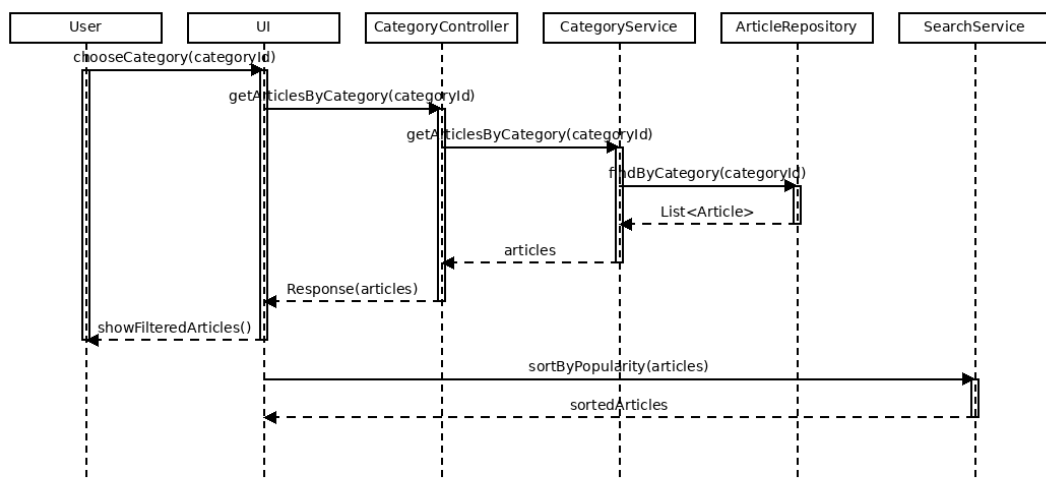
US-08 — Report Article



US-09 — Filter Articles •

فیلتر بر اساس دسته‌بندی مطابق Activity Diagram با CategoryController/CategoryService انجام می‌شود و سپس می‌توان مرتب‌سازی محبوبیت را انجام داد.

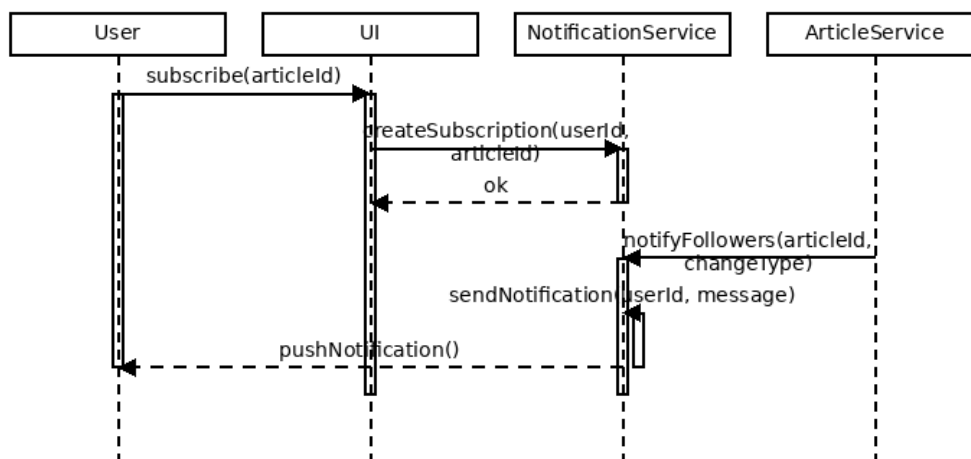
US-09 — Filter Articles



US-10 — Receive Change Notifications •

کاربر اشتراک اعلان را فعال می‌کند. بعد از تغییر مقاله، **ArticleService** تغییر را به **NotificationService** می‌دهد و اعلان برای کاربر ارسال می‌شود.

US-10 — Receive Change Notifications



8_ توضیحات :

در این فاز از پروژه برای رسم نمودار ها از [این ابزار](#) استفاده شده است که فایل های مربوطه برای کار با این ابزار به صورت txt. در فایل زیپ قرار گرفته شده است

