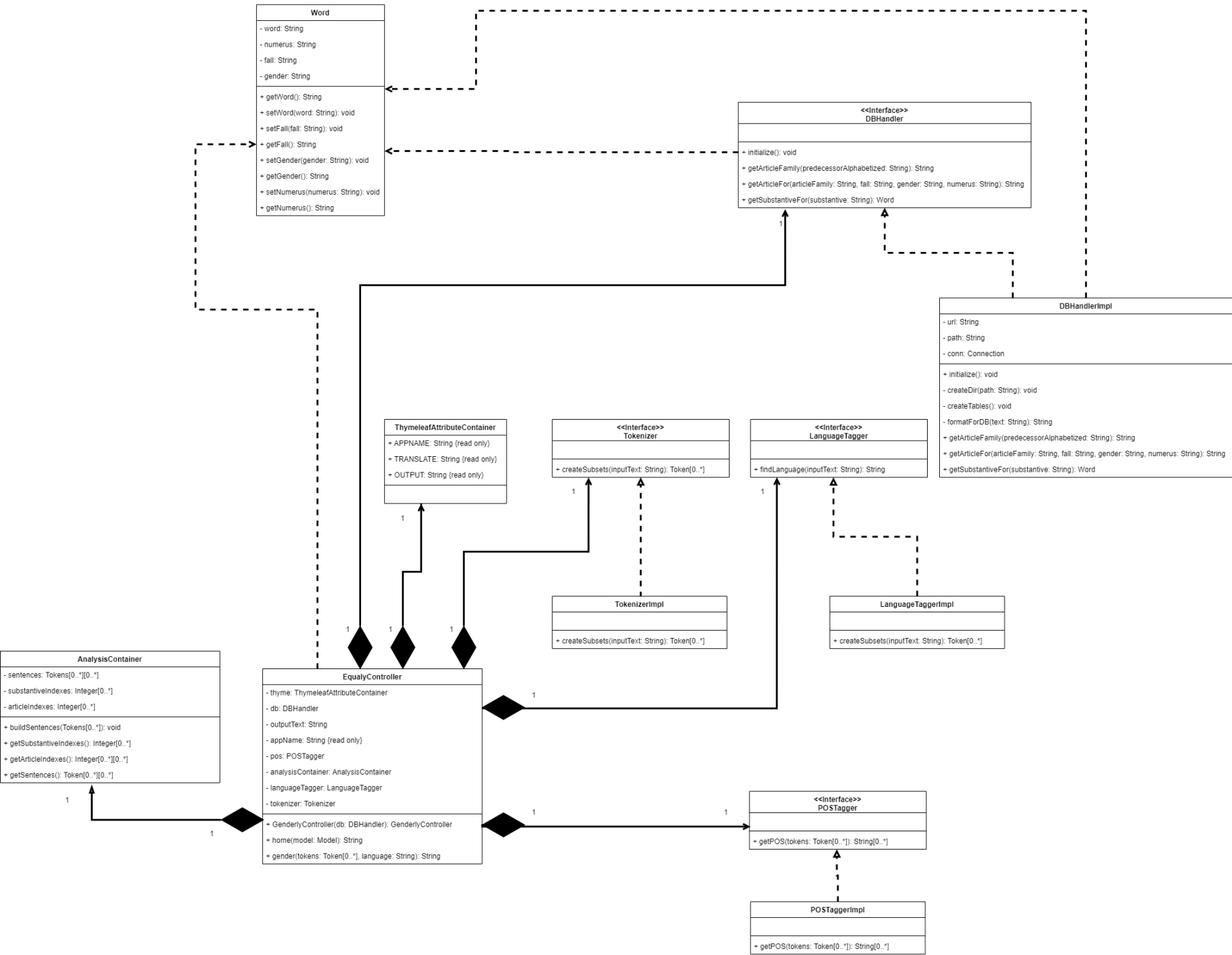


Equally UML-Klassendiagramm

Gender Your Texts Correctly.

Felix Jacobsen, Dennis Podkolsin, Marcus Koppelman





Gender Your Texts Correctly

Felix Jacobsen, Dennis Podkolsin, Marcus Koppelman

Das vorliegende UML-Klassendiagramm visualisiert die Klassen und Interfaces sowie internen Abhängigkeiten und Zusammenhänge der Software Equaly. Die Klassenstruktur wurde grundlegend nach dem Architekturmuster MVC entworfen und wird mit dem Framework Spring in Java realisiert. Ein *EqualyController* hält die Methode *home*, über die mittels Spring die Interaktion mit einer Weboberfläche stattfindet. Außerdem hält dieser Controller alle für den Gendering-Prozess relevanten Objekte.

Text trifft also über die Methode *home* aus dem HTML-, CSS- und JavaScript-Frontend im Backend ein und wird mittels der Methode *gender* unter – in Reihenfolge - Einbezug von Instanzen der Klassen *LanguageTagger*, *Tokenizer* und *POS-Tagger* zerlegt und analysiert. Es wird zunächst die Sprache festgestellt, dann wird der Text in Token/Sinnfragemente zerlegt. Anschließend werden für alle Sinnfragemente im POS-Tagger die Wortarten bestimmt.

Der durch diesen Anreicherungsprozess innerhalb der Methode *gender* mögliche Substitutionsprozess für genderbare Satzinhalte/Worte legt seine Zwischenergebnisse in einer *AnalysisContainer*-Instanz ab. Zwischenergebnisse treten zunächst bei der Lokalisierung und Substitution von Substantiven auf. Die Methode *gender* braucht die Ergebnisse dieser Substitution, denn nur die Sätze mit tatsächlich substituierten Substantiven müssen überhaupt auf möglicherweise zu substituierende Artikel untersucht werden. Das rechtfertigt die Existenz des *AnalysisContainers* und begründet den *DBHandler* als Realisierung des Models aus MVC. Eine Datenbank fungiert als Wörterbuch. In ihr wird für ein Wort sein Substituent gesucht.

Im UML-Diagramm sind die Pfeile z.B. für Kompositionen absichtlich an Interfaces und nicht an ihre Implementierung(en) gebunden. Das wurde zwecks Loose Coupling tatsächlich mittels Spring genau so realisiert. Eine Implementierung interagiert nur mit einem Interface, dessen Implementierung durch Spring im Hintergrund herausgesucht wird.

Wortanzahl: 250