

УДК 004.582

## РАЗРАБОТКА ВЕБ-АССИСТЕНТА ДЛЯ ВВОДА ЗАПРОСОВ С СИНТАКСИЧЕСКИМ АНАЛИЗАТОРОМ И ПОДСВЕТКОЙ СИНТАКСИСА НА СТОРОНЕ КЛИЕНТА.

Кузьминов И.В. (Санкт-Петербург, Университет ИТМО)  
Научный руководитель – степень, звание **Фамилия И.О.**  
(Полное название организации)

На данный момент существуют различные редакторы исходного кода, которые уже имеют встроенные анализаторы кода, подсвечивают синтаксис, позволяют гибко работать со структурой написанного кода (запроса) и выводят подсказки. Но когда речь заходит о поисковой строке веб-приложения, которая имеет поддержку какого-либо языка запросов, то скорее всего, в ней не будет подобных инструментов для работы с запросом. Целью данного исследования является разработка веб-ассистента, имеющего схожий функционал с редакторами исходного кода, применение которого для ввода запроса заметно упрощает читаемость введенного запроса, а также повышает производительность пользователя, который его использует.

**Введение.** Такой веб-ассистент особенно важен в тех приложениях, где строка поиска является одной из главных частей приложения. Например, в трекере задач, где строка поиска активно используется конечными пользователями для поиска и фильтрации задач, а сам компонент находится почти на всех «страницах» приложения.

Веб-ассистент для ввода запросов разрабатывался для трекера задач YouTrack компании JetBrains. Данный трекер имеет свой язык запросов, который содержит несколько операторов с различными приоритетами, унарные операторы, зарезервированные слова, поддерживает объединения.

**Основная часть.** Чтобы веб-ассистент был полезен, он должен подсвечивать синтаксис, указывать на ошибки в запросе, выводить подсказки, обращаться к серверу для более продуктивной работы.

Для реализации веб-ассистента необходимо использовать HTML элемент, который позволял бы встраивать в себя HTML разметку именно как разметку, а не как обычный текст, вследствие чего, нет возможность использовать элементы “input” или “textarea”. Поэтому было решено использовать элемент “div” с атрибутом “contenteditable” в значении “true”. Данный элемент имеет схожее поведение с “input”.

Пользователь будет использовать вышеуказанный элемент для ввода запроса. Поэтому требуется отслеживать все действия пользователя внутри данного элемента (ввод символов, клики, перемещение каретки, выделения и т.п.) и реагировать на эти действия. Например, для того чтобы подсветить запрос необходимо, чтобы после каждого введенного/удаленного символа запускался процесс анализа запроса и вставки новой HTML разметки в текущий компонент.

Веб-ассистента включает в себя следующие основные компоненты:

1. Лексический анализатор (лексер) – на первом этапе необходимо «прочитать» введенный запрос, чтобы понять какие символы используются, объединить цепочки символов в токены с указанием типа токена, с которыми в дальнейшем будет работать парсер, проверить запрос на наличие недопустимых символов.

2. Синтаксический анализатор (парсер) – после того, как запрос разбит на токены, требуется сделать его синтаксический анализ и построить дерево парсинга в соответствии с грамматикой используемого языка. Парсер – основной компонент веб-ассистента, после которого начинается работа не с самим запросом, а с построенным деревом парсинга. Дерево парсинга представляет собой абстрактную синтаксическую структуру, описывающую запрос.

При обходе дерева в глубину мы выполним запрос в правильном порядке с учетом приоритетов операторов (если таковые имеются), а также их особенностей (бинарные, унарные и т.п.).

3. Каретка – так как большинство популярных браузеров после вставки новой разметки в компонент сбрасывают каретку в начало или конец строки (в зависимости от конкретного браузера), реализован собственный компонент, который следит за положением каретки и корректно восстанавливает ее положение, а также работает с выделенным текстом.

4. Компонент подсветки – чтобы подсветить различные элементы запроса соответствующими цветами, нужно обойти дерево парсинга и составить HTML разметку с соответствующими стилями. Затем встроить ее в страницу.

5. Стек состояний – поскольку после вставки новой разметки в компонент, браузер «теряет» собственный стек состояний и не может отменить текущее действие, приходится следить за состоянием самостоятельно, не рассчитывая на помощь браузера. Для этого используются два стека “undo” и “redo” действий, которые хранят состояния компонента (запрос в текстовом виде, а также положение каретки).

6. Компонент подсказок – всплывающий роруп элемент, который выводит список подсказок из допустимых значений, либо релевантных значений с сервера (в случае, если компонент обращается к таковому).

7. Компонент работы со структурой – обрабатывает комбинации клавиш для работы со структурой запроса (выделение по синтаксическим частям и т.п.). Для работы данного компонента также используется дерево парсинга, полученное на этапе синтаксического анализа.

**Выводы.** В результате проделанной работы был реализован компонент на основе грамматики языка системы управления задачами YouTrack компании JetBrains. В настоящее время идет постепенное внедрение компонента в систему, некоторые части уже внедрены и доступны конечным пользователям.

Кузьминов И.В. (автор)

Подпись

Фамилия И.О. (научный руководитель)

Подпись