

Highcharts имеет в своем арсенале все самые известные типы графиков: Pie (разбиение круга на секторы), Column (отображение прогрессии столбцов), Spline (для разнородных метрик), Area или Areaspline (демонстрация эффекта от наложения метрик).

Следствие того что Highcharts использует технологию SVG является ряд достоинств, таких как: динамическое обновление данных – при изменении входной информации, например, получение обновленных данных от сервера, график видоизменяется; отключение отображения некоторых типов графиков с экрана благодаря возможности прикрепления событий; отображение подсказок, отображающих актуальную информацию о данных соответствующих данному участку графика.

Таким образом, библиотека для постройки графиков Highcharts представляет возможность создания различных вариаций графиков и настройки их представления. Highcharts является активно поддерживаемой библиотекой, что обеспечивает эффективность ее работы. Конечно, если необходим более гибкий инструмент для работы с графиками, можно создать свой собственный сервис, но его создание и поддержка потребует значительных затрат времени и сил.

**В.А. Ефремов (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.С. Давыдов**, канд. техн. наук, доцент

## **ИСПОЛЬЗОВАНИЕ ТАБЛИЧНЫХ МОДУЛЕЙ В ВЕБ-ПРОГРАММИРОВАНИИ НА ПРИМЕРЕ UI-GRID**

Статистические данные больших объемов удобно структурировать в виде таблиц, однако стандартные функциональные возможности таблиц не позволяют использовать их как полноценный механизм. Построение таблиц с помощью HTML тэгов является затратным по времени процессом, а результат не обладает достаточным функциональным набором, вследствие чего приходится самостоятельно создавать функции фильтрации данных, сортировки и так далее. Табличная верстка использовалась в html в качестве первого способа позиционирования элементов на странице, но позже она была вытеснена блочной версткой, а таблицы стали элементом представления статистической информации.

Для упрощения разработки табличных представлений можно использовать специальные табличные модули, например, UI-grid.

UI-grid является продолжением проекта Ng-grid. В данный момент, UI-grid не имеет официального релиза, но является более стабильной и функционально наполненной версией своего предка. Основные возможности данной библиотеки: Фильтрация данных, сортировка данных, разбиение контента на страницы, редактирование данных UI-grid требует наличия фреймворка Angular.js, и подключается к приложению, как обычный сервис. Организация представлений происходит с помощью директив «ui-grid», «ui-grid-edit», «ui-grid-cellnav» и других.

Обязательным является атрибут UI-grid, который принимает параметры таблицы. Сами же свойства должны содержать описания колонок. Для каждой колонки можно отдельно указать фильтры, а отображение поля ввода данных поиска достигается установкой флага «enableFiltering». Сортировка происходит автоматически в вариантах ASC, DESC и «нейтральном». В отличие от Ng-grid, данные сортируются по всей таблице, а не по одной странице.

Существует два вида разбиения данных на страницы в UI-grid. Первый способ – использование директивы «ui-grid-paging», которая добавляет интерфейс для разбиения данных по страницам, самой панели в таком случае библиотек не представляет. В таком случае проще всего использовать данную директиву вместе с bootstrap панелью для страниц и написанием собственных обработчиков событий. Второй способ – использование готовой панели при подключении директивы «ui-grid-pagination».

Для возможности редактирования данных в таблице, нужно в опциях колонки указать «editType» и, в зависимости от типа поля, указать дополнительные свойства, а в тэге таблицы указать атрибут «ui-grid-edit».

В опциях UI-grid есть функция «onRegisterApi», которая принимает API таблицы. С помощью данной функции можно настроить более сложные механизмы фильтрации и сортировки путем создания обработчиков событий конкретного типа действия. Например, имея на странице несколько таблиц, из которых одновременно отображается только одна, мы не должны создавать отдельную панель страниц для каждой таблицы. Достаточно создать единую панель, а в методе «onRegisterApi» для каждой таблицы инициализировать одну и ту же переменную собственным API. Метод «onRegisterApi» будет срабатывать только для активной таблицы.

Таким образом, библиотека UI-grid обладает всеми необходимыми возможностями для представления данных в виде таблиц и манипуляций этими данными.