

НЕЧЕТКАЯ ЛОГИКА В СИСТЕМАХ ПРОГНОЗИРОВАНИЯ ХРОНИЧЕСКИХ ЗАБОЛЕВАНИЙ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Зыщик М. А., Халецкий С. Д.

Лашкевич Е. М. – магистр технических наук, ассистент

Рассмотрены шаблоны проектирования и архитектура приложения системы прогнозирования развития хронического остеомиелита.

Остеомиелит (osteomyelitis, от греч. *osteon* – кость + *myelos* - костный мозг + *-itis*) – воспаление костного мозга, обычно распространяющееся на губчатое и компактное вещество кости и надкостницу [1]. В случае, если лечение начато поздно, острая форма течения заболевания может переходить в хроническую, т.е. может развиваться хронический остеомиелит. Особенно опасно это для детей, поскольку может приводить к деформации и укорочению кости. При этом на протяжении жизни пациента рецидивы и ремиссии постоянно сменяют друг друга, кроме того при хронической форме обязательно образование свища, не залечиваемого ни при каких условиях, из которого истекает гной. Таким образом, своевременное прогнозирование перехода развития заболевания в хроническую форму является актуальной задачей современной медицины.

Однако, как известно, прогнозирование развития заболевания – это всегда сложный процесс, основанный на личном опыте и знаниях врача. Для облегчения данной задачи в докладе предлагается создание системы прогнозирования развития хронического остеомиелита в виде программного приложения использующего в качестве инструмента получения вывода математический аппарат нечеткой логики.

Архитектура приложения. В качестве языка программирования для данной системы был выбран C#.

При разработке приложения будут использоваться три шаблона проектирования:

- MVVM (Model-View-View-Model) - используемая с целью разделения модели и её представления, что необходимо для изменения их отдельно друг от друга и удобства «связывания данных»;
- Singleton - гарантирующий, что в однопоточном приложении будет единственный экземпляр класса с глобальной точкой доступа;
- Facade - позволяющий скрыть сложность системы путем сведения всех возможных внешних вызовов к одному объекту, делегирующему их соответствующим объектам системы;

Шаблон MVVM (рисунок 1) имеет три основных компонента:

- модель, которая представляет бизнес-логику приложения,
- представление пользовательского интерфейса XAML,
- представление-модель, в котором содержится вся логика построения графического интерфейса и ссылка на модель, поэтому он выступает в качестве модели для представления.

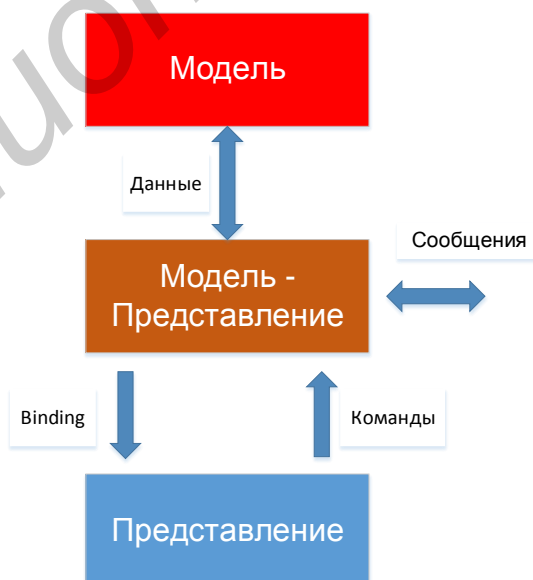


Рис. 1 – Реализация шаблона MVVM

Шаблон Singleton (рисунок 2) гарантирует, что у класса есть только один экземпляр, и предоставляет к нему глобальную точку доступа. Будет использоваться для хранения настроек пользователя и логирования его действий в системе.

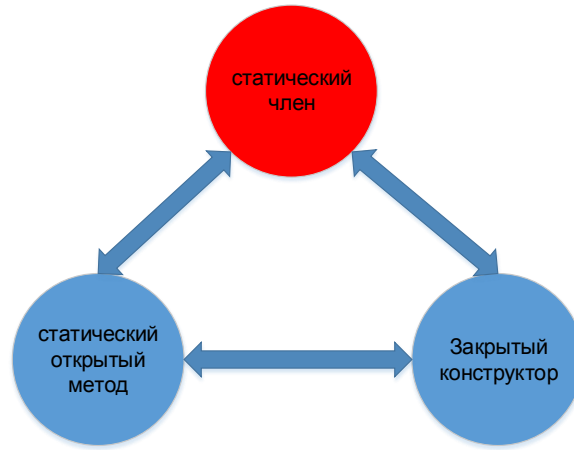


Рис. 2 – Реализация шаблона Singleton

Шаблон Facade (рисунок 3), который в будущем будет иметь имя DBDataAnalyzer, будет содержать анализаторы (фильтры) которые позволят искать информацию в базе знаний на основании указанных данных пользователя.

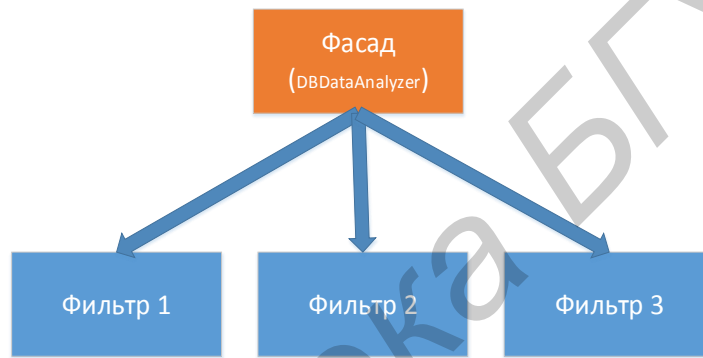


Рис. 3 – Реализация шаблона Facade.

Схема работы приложения представлена на рисунке 4

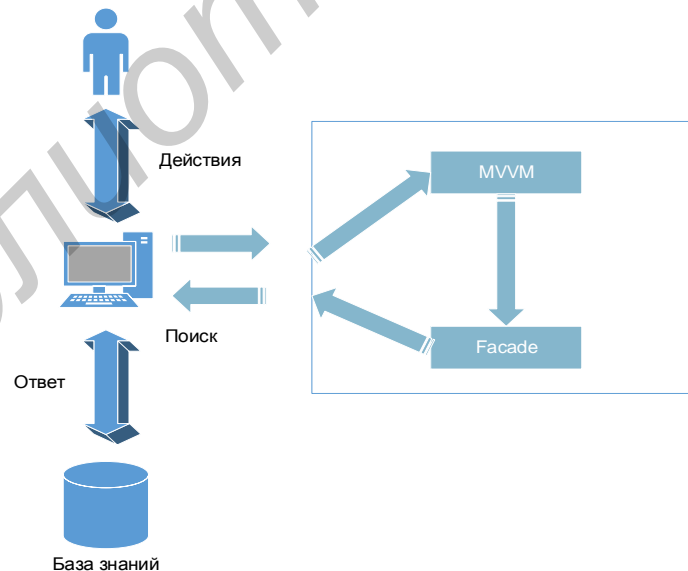


Рис. 4 – Работа приложения

Список использованных источников:

1. Острый гематогенный остеомиелит у детей. - БелМА... [Электронный ресурс]. – Режим доступа http://www.belmapo.by/downloads/dets_hirurgiy/prezentacija/ostrij_gematogennij.pdf. – Дата доступа 18.02.2014.

: