



*Рис. 2. Персонаж, созданный в Blender*

После создания модели ее нужно текстурировать, т.е. разукрасить персонажа, это достигается путем выделения нужной области и воспользоваться материалом или структурой. После этого приведем в движение нашего героя посредством «риггинга» (rigging). Данный процесс означает создание арматуры и связанных с нею ограничений. «Скиннингом» (skinning) называется присоединение сетки к арматуре таким образом, чтобы манипуляции с арматурой приводили к соответствующим деформациям сетки. Сетка должна следовать движениям арматуры приемлемым образом. В общих чертах это достигается путём назначения каждой кости влияния на движение определённых вершин [3].

На завершающем этапе анимации получившегося персонажа помещаем в игровой движок Unity для дальнейшего использования его в приложении или игре.

#### ЛИТЕРАТУРА

1. Anurag Ghosh. What is 3D Modeling? [Электронный ресурс]. — Режим доступа: \www/ URL: [http:// www.wisegeek.com/what-is-3d-modeling.htm#](http://www.wisegeek.com/what-is-3d-modeling.htm#).
2. 6 Industries that Use 3D Modeling Software [Электронный ресурс]. — Режим доступа: \www/ URL: <http://www.steves-digicams.com/knowledge-center/how-tos/video-software/6-in-dustries-that-use-3d-modeling-software.html#b>.
3. James Chronister. Blender Basics 4-rd edition, 2011. - 177 с.

#### ПРЕИМУЩЕСТВА ПРЕДМЕТНО-ОРИЕНТИРОВАННОГО ПРОЕКТИРОВАНИЯ

*К.М. Тюменцева*

*(г. Томск, Томский политехнический университет)*

*E-mail: kmt1@tpu.ru*

#### ADVANTAGES OF DOMAIN-DRIVEN DESIGN

*К.М. Tyumentseva*

*(Tomsk, Tomsk Polytechnic University)*

This article describes basic principles and rules of constructing domain model and advantages of domain-driven design.

**Keywords:** Domain-driven design, model-driven design, domain model, ubiquitous language.

При разработке программного обеспечения встречаешься с различными трудностями. Основная естественная преграда - это предметная область, в которой существует данная проблема.

Любое программное обеспечение имеет применение в той или иной сфере деятельности или области интересов. Приведем пример, наиболее связанный с реальным миром: онлайн покупка билетов на самолет. То есть предметная область – это такая область знаний, в которой пользователь использует ПО.

Все чаще встречаются команды разработчиков, занимающиеся проектированием ПО по модели предметной области. Предметно-ориентированное проектирование – это набор правил и схем, позволяющих создавать программные абстракции (модель предметной области), отображающие бизнес-логику.

Для реализации ПО из предметной области, которая неизвестна разработчику, ему необходимо восполнить недостаток, но не читая многостраничные и малопонятные книги, научные статьи, поскольку это даст расплывчатое представление. Инструментом для преодоления этой трудности служит модель, которая строится из намеренно упрощенных и строго отобранных знаний. Если модель позволяет сосредоточиться на проблеме, то значит она построена правильно [1].

Поскольку модель концентрирует знания вокруг конкретной проблемы, это ограничивает применение модели в определенной области деятельности – контексте. Контексты имеют границы, которые окружают ту или иную модель [2].

Модель и архитектура программы тесно связаны. Это еще одна причина, помимо сложности предметной области, почему программисты включаются в процесс построения модели.

В проектах, в которые нет модели предметной области (где архитектура никак не связана с моделью предметной области), используется аналитическая модель, которую разрабатывают не учитывая потребностей проектирования приложения. Проектирование по модели (Model-Driven design) ликвидирует разрыв между аналитической моделью и архитектурой, так как строится такая модель, которая служит обеим целям [3].

Сравнения с каскадной методологией, где специалисты предметной области доставляли информацию аналитикам, те ее перерабатывали и предоставляли разработчикам. Такой подход однонаправленный, программа решает поставленную задачу, но имеет меньше возможностей к успеху.

Коммуникация и язык: Обычно требования описываются в виде пользовательской истории в формате: Я как [роль], Хочу [функция], Чтобы [цель]. Рассмотрим пример: Я как консультант, хочу контролировать одобрение полисов, чтобы заключать надежные соглашения и отклонять рискованные. Хорошо понимаете, что здесь имеется в виду? Проекты очень часто сталкиваются с подобными языковыми препятствиями: специалисты говорят на своем профессиональном языке, программисты – на техническом. Тогда аналитикам приходится помогать в понимании друг друга внутри команды. Что может привести к несовместимости программных частей.

Подход DDD предлагает формировать единый язык, который используется не только для описания объектов модели, но и функциональных возможностей, сценариев использования и приемочных тестов. Язык эволюционирует вместе с моделью, то есть все изменения в языке отображаются изменениями в коде: переименование, классов, методов и переменных. Единый язык позволяет избежать рефакторинга и упростить коммуникацию между разработчиками и специалистами предметной области. Так же используя единый язык всей командой возможно выявить слабые места в модели и противоречия в коде.



Рис.1 – Образование единого языка

Документация в данном подходе служит дополнением к коду и устным обсуждениям, то есть рассказывает, как используется модель. Подход сокращает объем документации и позволяет формировать ее в упрощенной форме. Так как требования в гибких методологиях постоянно изменяются, то актуальность документации можно проверить при помощи единого языка.

Предметно-ориентированное проектирование позволяет выработать индивидуальный подход к реализации проекта. Выбрать наиболее подходящие методы, при этом не исключая использование общепринятых подходов к разработке ПО. Таким образом проектирование сложной бизнес-логики предметной области в виде модели способствует качественной проработки технической инфраструктуры и успешной реализации проекта.

#### ЛИТЕРАТУРА

1. Eric Evanc Domain-driven design Tackling Complexity in the Heart of Software, April 15, 2003. – 359.
2. Domain-driven design.Context. URL: <http://blog.sectorit.net/domain-driven-design/1641>
3. Введение в проблемно-ориентированное проектирование. URL: <https://msdn.microsoft.com/ru-ru/magazine/dd419654.aspx?id0090088>

#### ОСОБЕННОСТИ ВИЗУАЛИЗАЦИИ ТРАНСПОРТНЫХ ПОТОКОВ В УСЛОВИЯХ ГОРОДА

*М.Е. Хомеченко, Д.Д. Исунц*

*Научный руководитель: Дорофеев В. А.*

*Национальный Исследовательский Томский Политехнический Университет*

#### SOME ASPECTS OF TRAFFIC STREAMS VISUALIZATION IN THE CITY ENVIRONMENT

In today's world there is a global popularization of urban transport. Transport system evolves continuously. In this regard, right strategy of traffic control question is very important. The article describes the traffic flow models that can be used for modeling close-to-real traffic situations. The project uses Unity 3D game engine and C# programming language for urban environment visualization; it will be used for modelling one of the traffic flow models. Whole model will