

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем
Направление 09.04.04 – Программная инженерия
Магистерская программа Управление разработкой программного обеспечения

ДОПУСТИТЬ К ЗАЩИТЕ

Зав. кафедрой

_____ А.В. Бушманов

« _____ » _____ 2017г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему: Компьютерная реконструкция Цусимского сражения

Исполнитель

магистрант группы 557-ом

(подпись, дата)

А.В. Сороговец

Руководитель

доцент, канд. физ.-мат. наук

(подпись, дата)

В.В. Еремина

Руководитель

магистерской программы
профессор, доктор техн. наук

(подпись, дата)

Е.Л. Еремин

Нормоконтроль

доцент, канд. физ.-мат. наук

(подпись, дата)

В.В. Еремина

Рецензент

доцент, канд. техн. наук

(подпись, дата)

А.Н. Рыбалев

Рецензент

доцент, канд. техн. наук

(подпись, дата)

М.Д. Штыкин

Благовещенск 2017

РЕФЕРАТ

Магистерская диссертация содержит – 74 с., 41 рисунок, 9 таблиц, 3 приложения, 54 источника. Структура диссертации состоит из трех разделов: «Обзор современного состояния проблемы компьютерных реконструкций исторических сражений», «Алгоритмическое и программное обеспечение», «Практическая реализация компьютерной реконструкции».

3D - МОДЕЛИРОВАНИЕ, ПРОЕКТИРОВАНИЕ, БАЗА ДАННЫХ, ВИЗУАЛИЗАТОР, ПОЛИГОНАЛЬНОЕ МОДЕЛИРОВАНИЕ, 3D-МОДЕЛЬ, ЦУСИМСКОЕ СРАЖЕНИЕ, РУССКО-ЯПОНСКАЯ ВОЙНА, ВИРТУАЛЬНАЯ РЕКОНСТРУКЦИЯ, ИСТОРИКО-КУЛЬТУРНОЕ НАСЛЕДИЕ, КОРАБЛЬ, КОМПЬЮТЕРНАЯ ГРАФИКА, АЛГОРИТМ, РЕКОНСТРУКЦИЯ БИТВЫ, ВОЕННО-ИСТОРИЧЕСКИЙ ТУРИЗМ, МЕТОДЫ ИСТОРИЧЕСКОГО ИССЛЕДОВАНИЯ

Актуальность исследования состоит в том, что в настоящее время наблюдается интенсивное применение трехмерной графики в исторических исследованиях. Наиболее актуальным и бурно развивающимся научно-практическим направлением данной задачи выступает разработка виртуальных реконструкций. Таким образом, восстановления объектов и событий историко-культурного наследия посредством трехмерных технологий является достаточно актуальной задачей.

Целью работы является повышение интереса к объектам историко-культурным наследия, путем создания виртуальной реконструкции Цусимского сражения.

Научная новизна основных результатов работы: для боевых кораблей про-

					<i>ВКР.155518.09.04.04.ПЗ</i>			
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>				
<i>Разраб.</i>		<i>Сороговец А.В.</i>			КОМПЬЮТЕРНАЯ РЕКОНСТРУКЦИЯ ЦУСИМСКОГО СРАЖЕНИЯ	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Пров.</i>		<i>Еремина В.В.</i>				У	2	74
<i>Н. контр.</i>		<i>Еремина В.В.</i>				<i>АмГУ кафедра ИУС</i>		
<i>Зав. каф.</i>		<i>Бушманов А.В.</i>						

тывоборствующих сторон Цусимского сражения созданы трехмерные модели, учитывающие их реальные пропорции и максимальную детализацию реально существовавших прототипов; предложен метод построения редактора сценариев, осуществляющий создание сцены боя; разработан программный продукт, представляющий собой компьютерную реконструкцию Цусимского сражения.

Практическая ценность диссертационной работы: разработанный программный продукт позволяет достичь эффекта проникновения человека в историческую среду, что в свою очередь способно вызывать эмоциональный отклик у зрителя. Благодаря зрелищности, наглядности и высокому уровню детализации, виртуальная реконструкция поможет привлечь внимание пользователей всех возрастов и любого уровня образования к вопросам истории, а также может использоваться в образовательной деятельности, в задачах развития историко-культурного туризма.

Итоговая система получилась универсальной, что позволяет легко адаптировать ее для применения в других предметных областях.

Защищаемые положения:

- 1) трехмерные модели боевых кораблей противоборствующих сторон Цусимского сражения;
- 2) метод построения редактора сценариев, осуществляющий создание сцены боя и позволяющий осуществление нелинейного воспроизведения сценария;
- 3) программный продукт, представляющий собой компьютерную реконструкцию Цусимского сражения.

Основные результаты проведенного квалификационного исследования опубликованы в двух работах, среди которых одна журнальная статья [35] и одни тезисы доклада на научной конференции [36].

					<i>ВКР.155518.09.04.04.ПЗ</i>			
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>				
<i>Разраб.</i>		<i>Сороговец А.В.</i>			<i>КОМПЬЮТЕРНАЯ РЕКОНСТРУКЦИЯ ЦУСИМСКОГО СРАЖЕНИЯ</i>	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Пров.</i>		<i>Еремина В.В.</i>				У	3	74
<i>Н. контр.</i>		<i>Еремина В.В.</i>				<i>АмГУ кафедра ИУС</i>		
<i>Зав. каф.</i>		<i>Бушманов А.В.</i>						

СОДЕРЖАНИЕ

Введение	7
1 Обзор современного состояния проблемы компьютерных реконструкций исторических сражений	10
1.1 Особенности виртуальных исторических реконструкций	10
1.2 Опыт разработки аналогов системы виртуальной реконструкции	15
1.3 Цусимское сражение и состав противоборствующих флотов	18
2 Алгоритмическое и программное обеспечение	35
2.1 Обоснование выбора программы для 3D моделирования	35
2.2 Обоснование выбора визуализатора	40
2.3 Обоснование выбора СУБД	44
3 Практическая реализация компьютерной реконструкции	48
3.1 Создание 3D-моделей кораблей	48
3.1.1 Описание инструментов и этапов моделирования	48
3.1.2 Создание полигональной модели	49
3.1.3 Реализация синтетической текстуры	52
3.2 Создание редактора сценариев	54
3.3 Проектирование интерфейса программного модуля	58
3.4 Проектирование базы данных	62
3.5 Руководство пользователя	66
Заключение	69
Библиографический список	70
Приложение А Карта боевых действий Цусимского сражения	75
Приложение Б Листинг интерпретатора сценариев	76
Приложение В Логическая и физическая модели базы данных	79

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей диссертационной работе использованы ссылки на следующие стандарты и нормативные документы:

ГОСТ 2.104-68 ЕСКД Основные надписи.

ГОСТ 2.105-95 ЕСКД Общие требования к текстовым документам.

ГОСТ 2.111-68 ЕСКД Нормоконтроль.

ГОСТ 7.1-2003 Библиографическое описание документа. Общие требования и правила составления.

ГОСТ 19.201-78 ЕСПД Техническое задание. Требования к содержанию и оформлению.

ГОСТ 19.401-78 ЕСПД Текст программы. Требования к содержанию и оформлению.

ГОСТ 19.404-79 ЕСПД Пояснительная записка. Требования к содержанию и оформлению.

ГОСТ 24.206-80 Требования к содержанию документов по техническому обеспечению.

ГОСТ 24.207-80 Требования к содержанию документов по программному обеспечению.

					<i>ВКР.155518.09.04.04.ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		5

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

RTS – Real Time Strategy;

БД – база данных;

СУБД – система управления базами данных.

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		6

ВВЕДЕНИЕ

Широкое развитие компьютерных и мультимедийных технологий способствовало появлению новых форматов продвижения и популяризации исторической информации, ее обработки, новых методов исследования. Все большее распространение приобретает применение трёхмерных технологий для возможности сохранения в цифровом виде объектов историко-культурного наследия и обеспечения к ним свободного доступа не только исследователей, но также самой широкой пользовательской аудитории.

Применение трехмерной графики в исторических исследованиях позволяет визуализировать внешний облик исследуемого объекта, воссоздать культуру той или иной исторической эпохи, максимально правдоподобно передать суть исторического события.

Одним из наиболее актуальных и бурно развивающихся научно-практических направлений восстановления объектов историко-культурного наследия выступает разработка виртуальных реконструкций, основанных на использовании технологий 3D-моделирования. Применение виртуальных реконструкций дает возможность эмоционального погружения в историческое событие, позволяет увидеть его в первоизданном виде, увеличивает скорость передачи информации и повышает уровень ее понимания [34]. Полученные результаты обладают историко-культурной значимостью и могут эффективно использоваться в образовательном процессе.

Особый интерес вызывают виртуальные военно-исторические реконструкции, воссоздающие ход боевых действий и сражений. При этом виртуальные реконструкции выступают неотъемлемым элементом накопления и сохранения культурных ценностей государства, усиливают интерес к истокам исторической и культурной идентичности, способствуют формированию патриотических чувств. Утраты исторических ценностей невозможны и необратимы, что выдвигает на повестку дня вопрос о необходимости сохранения и донесения достоверных знаний о тех или иных значимых исторических событиях.

					<i>ВКР.155518.09.04.04.ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		7

Тем самым, виртуальная реконструкция Цусимского сражения, решившего исход русско-японской войны, является важнейшей задачей сохранения культурного наследия в общественном сознании.

Экспонирование исторической компьютерной реконструкции позволит на качественно новом уровне представить на экранах монитора максимально правдоподобное историческое событие, раскрывающее ход Цусимского сражения времен русско-японской войны.

Объектом исследования диссертационной работы выступает крупнейшее в эпохе додредноутного броненосного флота Цусимское сражение.

Целью диссертационной работы является повышение интереса к объектам историко-культурного наследия, путем создания виртуальной реконструкции Цусимского сражения.

Научная новизна результатов:

- 1) для боевых кораблей противоборствующих сторон Цусимского сражения созданы трехмерные модели, учитывающие их реальные пропорции и максимальную детализацию реально существовавших прототипов;
- 2) предложен метод построения редактора сценариев, осуществляющий создание сцены боя;
- 3) разработан программный продукт, представляющий собой компьютерную реконструкцию Цусимского сражения.

Практическое значение и реализация результатов работ.

Разработанный программный продукт, представляющий собой интерактивный фильм в формате виртуальной реальности, позволяет достичь эффекта проникновения человека (пользователя системы) в историческую среду, что в свою очередь способно вызывать эмоциональный отклик у зрителя. Благодаря зрелищности, наглядности и высокому уровню детализации, виртуальная реконструкция поможет привлечь внимание пользователей всех возрастов и любого уровня образования к вопросам истории, а также может использоваться в образовательной деятельности, в задачах развития историко-культурного туризма.

Итоговая система получилась универсальной, что позволяет легко адаптировать ее для применения в других предметных областях.

Положения, выносимые на защиту:

1) трехмерные модели боевых кораблей противоборствующих сторон Цусимского сражения;

2) метод построения редактора сценариев, осуществляющий создание сцены боя и позволяющий осуществление нелинейного воспроизведения сценария;

3) программный продукт, представляющий собой компьютерную реконструкцию Цусимского сражения.

Новизна и значимость технических решений подтверждена публикациями в научных изданиях.

Основные результаты диссертации докладывались и обсуждались на: XXV научной практической конференции АмГУ «Дни науки» – 2016 (14 мая 2016 года), XVII региональной научно-практической конференции «Молодёжь XXI века: шаг в будущее» (24 мая 2016 года), XXVI научной практической конференции АмГУ «Дни науки» – 2017 (20 мая 2017 года), XVIII региональной научно-практической конференции «Молодёжь XXI века: шаг в будущее» (18 мая 2017 года).

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		9

1 ОБЗОР СОВРЕМЕННОГО СОСТОЯНИЯ ПРОБЛЕМЫ КОМПЬЮТЕРНЫХ РЕКОНСТРУКЦИЙ ИСТОРИЧЕСКИХ СРАЖЕНИЙ

1.1 Особенности виртуальных исторических реконструкций

Способы работы историков с военно-историческими архивными документами, фотографиями и чертежами, а также историческим материалом формировались в течение долгого времени, методика усовершенствовалась, и, вследствие этого, инструментарий историка тоже существенно видоизменился. В настоящее время, наиболее перспективным из современных методов исторического исследования заслужено можно считать трехмерное моделирование [18].

Сущность технологии виртуальной реальности, как она используется в работе Жеребятъева Д. И.[12], описывается как программное обеспечение, применяемое для создания виртуальных миров, в состав которого входят технологии трехмерного моделирования и их результат – виртуальный интерактивный мир. Важной особенностью технологий виртуальной реальности выступает наличие интерактивности и погружения, благодаря чему пользователь способен перемещаться и взаимодействовать с объектами в трехмерном пространстве. Под понятием технологий 3D-моделирования зачастую применяется обозначение инструментов создания виртуальных трехмерных моделей.

Компьютерные технологии послужили катализатором развития моделирования. Вдобавок к математическому моделированию процессов истории в середине 90 годов 20 века, появляется новый вид моделирования – виртуальная реконструкция исторических событий [11], выступающая способом хранения и проявления значимости исторических событий, проверки исторических идей и суждений, их уточнения и создания новых. Так, на основе практической реализации какой-либо структуры (трехмерной графической, математической модели) появляются новые теоретические результаты, и наоборот – созданная 3D-модель обличает недостатки в существующих знаниях и технологиях, запуская новые практические разработки.

В отличие от традиционных инструментов разработки исторических ре-

					<i>ВКР.155518.09.04.04.ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		10

конструкций, таких как чертежи, эскизы и макеты, информационные технологии позволяют быстро дополнять и изменять полученные модели. Без трудоемкого перерисовывания появляется возможность редактирования параметров рельефа, изменения формы и материала исторического здания, моделирования условий освещения и природных явления и т.п.

Значимость внедрения технологий 3D-моделирования в искусствоведческие исследования отметил руководитель отдела культурного наследия Генеральной дирекции по вопросам информационного общества Европейской Комиссии Бернанд Смит в своем докладе [32], выделив важность применения технологий трехмерного моделирования для сохранения и воссоздания памятников культуры. Следственно, трехмерные технологии позиционируются современными исследователями как многофункциональный инструментарий для исследования и представления исторической информации и требуют организованного внедрения на основе соизмеримых методик и стандартов.

Следует заметить, что в данное время технологии трехмерного моделирования вступили в новый этап формирования исторического исследования, и выступают как метод изучения и обработки исторического материала [41]. Преодолев путь от единичных разработок трехмерных моделей до профессиональной обработки данных коллективом ученых-исследователей, технологии трехмерного моделирования становится неотъемлемой частью исторической науки. Действительно, результат труда приобретает статус исторического исследования.

Хотелось бы отметить, что на данный момент на научном поприще еще не сформировано полноценной типологии компьютерных исторических реконструкции, охватывающей все направления процесса трехмерного моделирования [3]. Условно трехмерные исторические реконструкции можно разделить на несколько групп, среди которых:

1. Реконструкция архаичных и древних городов в целях исследования особенностей создания и становления городской инфраструктуры, восстановления и реконструкции утраченных памятников архитектуры и культуры. Это

самый многочисленный тип исторической реконструкции по масштабу и количеству реконструируемых моделей. Он может включать в себя весь нижеперечисленный перечень реконструкций исторических предметов быта: построек и зданий, ландшафта, различных моделей интерьера помещений, людей, костюмов. В качестве примера такой реконструкции можно привести проект виртуальной реконструкции древнего города Рима, в ходе которого были восстановлены постройки города времен императора Константина I и улицы с его населением[54]. Работы по воссозданию виртуальных интерактивных моделей городов являются очень трудоёмкими и требуют больших затрат, а реконструкции внутреннего пространства зданий в данной типологии исторических моделей встречается довольно редко.

2. Пространственная реконструкция ландшафта имеет большое, ключевое значение и позволяет изучать особенности той или иной местности [6]. Помимо этого, она дает возможность восстановить в дальнейшем исторические события, например, сражения, используя модель ландшафта. Рассмотрение ландшафта позволяет симитировать исторический бой на местности, а также провести анализ расстановки войск и тактики полководцев. Еще одним важным фактором моделирования ландшафта является возможность рассмотреть широкий круг деталей и задач исторического исследования, таких как влияние и роль природно-географического фактора на конкретное историческое событие.

3. Реконструкция комплекса построек и в отдельности зданий. Здесь, трехмерной реконструкции подвергаются различные монастыри, соборы, церкви, часовни, дворцы, а также отдельные здания, обладающие культурной спецификой и имеющие большое историческое значение. На этапе создания трехмерной модели постройки моделируется не только внешний облик здания, но и его внутренний интерьер, что является весьма значимым для осуществления задач сохранения культурного наследия.

4. Отдельным видом в рамках типологии виртуальных исторических реконструкций выступает историческое реконструирование предметов быта, вооружения, археологических артефактов и находок и т.д. Работы в этой области

					<i>ВКР.155518.09.04.04.ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		12

ведутся в большом объёме и весьма успешно. Здесь задействованы, как историки и археологи в процессе оцифровки археологических находок и превращения их в трехмерные модели, так и музееведы в процессе оцифровки фондов музея.

5. Реконструкция интерьеров комнат исторических сооружений, виртуальные музеи. Технологии трехмерного моделирования способны воссоздать целые архитектурные комплексы, отражающие быт и культуру прошлых утраченных памятников архитектуры, части которых разбросаны по различным музеям. Так, например, скульптуры Парфенона, разбросанные по музеям Европы, в первый воссоединяются в рамках виртуальной реконструкции, реализацией которой сейчас занимается Калифорнийский институт креативных технологий [15].

6. Одним из наиболее важных видов исторических реконструкций выступает трехмерная реконструкция исторических битв и событий, в ходе которой происходит воссоздание в соответствии с планом отдельные аспекты некоторого исторического события, или последовательности событий. Возможно, трактовать это как желание сохранить и передать для будущих поколений многообразные события в жизни, как отдельного человека, так и целого общества или всего государства. Кроме того, можно предположить, что это один из способов понять, почему именно так определяется история для ее непосредственных участников и, что может ожидать их в ближайшем будущем.

В настоящее время положение дел реконструкции исторических событий и памятников истории характеризуется заметной активностью специалистов моделирования исторических объектов, как в игровом аспекте, так и на профессиональном уровне в области исторического исследования. Создаются исследовательские группы, лаборатории и институты, занимающиеся на практике разработкой виртуальных реконструкций памятников культуры, существующих до сих пор или когда-то существовавших, но не сохранившихся.

В конечном счете, в данное время, в мире реализуется огромное количество проектов в области сохранения историко-культурного наследия, в которых реализовано моделирование всех перечисленных типов виртуальных рекон-

струкций. Большая часть таких проектов создаются коллективами западных частных организаций и университетов, наиболее известные из которых – это Digital Design Research Group (США), Knauf (Германия), LAND Lab Pastscapes 3D Archaeological Reconstruction Company (Англия) и другие [24].

В России реализация таких проектов развивается более медленно, и сосредоточена в основном в западной части страны. Тем не менее, существуют проекты, например, коллективов Сибирского федерального университета, Алтайского университета, Института истории материальной культуры РАН, Тамбовского государственного университета, МГУ и других [40].

Разработка виртуальных реконструкций весьма нетривиальная задача и требует больших интеллектуальных и временных затрат. Условно этот процесс можно разбить на несколько этапов:

Первый этап – постановка задачи исторической реконструкции.

Вторым этапом реконструкции выступает определение круга доступных географических, изобразительных и описательных источников, т.е. формирование базы, на основе которой будет происходить построение трехмерных моделей. Существует целый ряд проблем, связанных с особенностями источников, самый основной из них – это недостаток необходимого достоверного материала [13].

Третьим этапом является верификация найденных источников, в ходе которого происходит проверка все собранных источников на достоверность и непротиворечивость. Анализ картографических и описательных источников является весьма трудоемким этапом и требует большого количества времени.

Четвертый этап виртуальной реконструкции заключается в построении трёхмерных моделей изучаемых объектов с полной детализацией [7, 8]. На этом этапе происходит полигональное моделирование объектов при помощи редактирования сетки объекта, состоящей из полигонов (четырёхугольников, треугольников или других многоугольников). Далее происходит текстурирование трехмерных объектов, где поверхностям трехмерных моделей придают вид реальных материалов для достижения максимальной реалистичности. Происхо-

дит проецирование текстур на поверхность модели в соответствии с картой UV-координат, где каждой вершине объекта ставится в соответствие определённая координата на двумерном пространстве текстуры. Модели приобретают вид различных материалов: дерева, металла, пластика. Для этого в любой программе трехмерного моделирования существуют редакторы материалов, в которых есть готовые наборы материалов или с помощью которых можно разработать собственные материалы. Также, значимой частью организации трехмерной сцены для виртуальной реконструкции является моделирование ландшафта.

Заключительным этапом реконструкции выступает построение интерактивной системы навигации пользователя в трехмерном пространстве. При этом процесс визуализации должен быть достаточно эффективным для обеспечения приемлемого для пользователя уровня интерактивности, если этого требуют задачи проекта.

В целом, эффективность использования технологий трехмерного моделирования в исследованиях должна оцениваться при решении конкретных задач пространственного моделирования объектов с учётом соответствующих особенностей.

1.2 Опыт разработки аналогов системы виртуальной реконструкции

Современные ученые подчёркивают, что одним из наиболее значимых по своим научным и социальным последствиям мероприятий с применением современных информационных технологий является виртуальная реконструкция исторических событий [33].

Компьютерная реконструкция исторических событий выступает обособленным видом реконструкций в типологии визуальных исторических моделей.

Чаще всего реконструкции исторических событий реализуются группой разработчиков при поддержке историков для игровой индустрии и научных фильмов. Популярным жанром таких игр является RTS (Real Time Strategy) или Action. Результаты исследований, выполняемых в этой области, крайне разнообразны и предоставляют возможность рассмотреть ход исторических событий, например сражений, от третьего лица, непосредственно принять участие в них

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		15

или симитировать возможные варианты исхода исторического действия.

Интересный взгляд на исторические компьютерные игры, как на способ моделирования исторической информации, изложен Яблоковым К.В в своей статье [48]. Автор выделяет лидером в этом направлении компанию History Channel, занимающуюся активным внедрением трехмерных проектов игровых симуляторов в сферу игровой индустрии, в основе которых находятся виртуальные исторические реконструкции. Судить об исторической значимости данных проектов, выпущенных под маркой популярного исторического телеканала, можно с определенной долей условности. В качестве примера приводится проект «Великие битвы Рима», являющийся современной RTS-стратегией, основанной на достоверных материалах и точной симуляции реально происходящих сражений [49].

Основой таких RTS стратегий выступает математическая модель [46]. Однако, как хорошо бы она ни была разработана, не возможно в полной мере учитывать все влияющие на ход боя факторы. Если поподробнее разобраться с данной технологией поведения моделей, к примеру, воинов, все они ведут себя по заранее описанному алгоритму, который, к сожалению, не совершенен. В нем не хватает многих элементов: например, компьютерным моделям не характерен страх, так как, для этого нужно создать систему виртуального разума, что является довольно сложной задачей и на данный момент по силам только единичным разработчикам. Исходя из вышесказанного, можно сделать вывод, что такая модель, где пользователь может оказывать влияние на процесс происходящего и создавать отклонения в ту или иную сторону, не несет исторической достоверности. Моделирование исторических событий, основанное на использовании программных движков RTS, в историческом анализе удачно применяется в ходе построения линейных цепочек развития событий и находит свое отражение в исторических научно-популярных фильмах.

На протяжении последних пяти лет за границей отмечается тенденция создания больших образовательных ресурсов по истории с компонентами виртуальной реальности, нацеленных на массовую аудиторию, а также на учебные

заведения и музеи. Главной целью проекта является популяризация исторических знаний. Слушателям представляются результаты исследований в виде научных фильмов с элементами трехмерного виртуального объекта (постройки, здания, предмета быта) или в виде интерактивной трехмерной реконструкции. Ведущими каналами в создании таких фильмов выступают американские каналы History Channel и National Geographic, которые активно сотрудничают с компаниями разработчиков, специализирующихся на оказании услуг в сфере трехмерного моделирования [1].

Индустрия компьютерных игр, ориентированная на массовую аудиторию, устремленная на организацию впечатляющих проектов, касающихся исторических событий: походы Александра Македонского, Римские войны, Средневековые, походы Наполеона Бонапарта, Первая и Вторая мировые войны и другие [30]. Статистические данные свидетельствуют, что популярность компьютерных игр неуклонно растет, а область их применения постоянно расширяется. В процессе создания трехмерных игр с историческим сюжетом, разработчики воссоздают рельеф местности, архитектурные постройки, одежду, вооружение, методы ведения войны, культуру и другие аспекты. Иногда в разработке компьютерных игр принимали участие профессиональные историки, как например, это было в проекте Assassin creed, в котором были приглашены три специалиста по истории, связанные с изучением истории Иерусалима в эпоху средневековья. К сожалению, на практике такое случается не часто. Результаты, достигаемые в области разработки компьютерных игр, поражают по своим масштабам и качеству детализации, но, к сожалению, к исторической достоверности они имеют опосредованное значение.

Применения виртуальных технологий в исторической среде ведутся не только за рубежом, но и в России. В Санкт-Петербурге крупнейшая компания «Невский баталист», занимающаяся созданием историко-художественных экспозиций, разрабатывает трехмерные панорамы исторически значимых сражений. В число выполненных проектов входят реконструкции битвы за Берлин, посвященная штурму Берлина во времена Великой Отечественной войны, и

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		17

битвы за Москву, реконструирующей момент сражения на знаменитом Бородинском поле. На сайте компании имеется возможность совершения виртуального тура по трехмерной панораме. Разработок, ведущихся в области реконструирования морских сражений, у данной компании нами найдено не было.

Отдельно стоит сказать о разработках касающихся военных морских сражений времен русско-японской войны. Подавляющее большинство представленных 3D-моделей кораблей и морских сражений охватывают период с 1920 г по настоящее время, а разработки периода русско-японской войны практически полностью отсутствуют. Наиболее полное собрание 3D-моделей кораблей представлено в World of Warships — клиентской массовой многопользовательской компьютерной онлайн-игре в реальном времени в жанре симулятора морских сражений, однако это не образовательный ресурс, кроме того, игра не охватывает исследуемый период. Таким образом, реальных конкурентов у проектируемого программного продукта фактически нет.

1.3 Цусимское сражение и состав противоборствующих флотов

Русско-японская война (27 января 1904 – 23 августа 1905) – война между Российской и Японской империями за контроль над Маньчжурией и Кореей. Стала – после перерыва в несколько десятков лет – первой большой войной с применением новейшего оружия: дальнобойной артиллерии, броненосцев, миноносцев [16, 47].

Завершающим этапом похода второй Тихоокеанской эскадры на Дальний Восток явилось Цусимское сражение 14 мая 1905 года в Цусимском проливе, являющиеся объектом реконструкции [4, 31]. Исходными данными для создания реконструкции служили сохранившиеся планы боя, а также записи судовых журналов и воспоминания очевидцев.

Главной целью русского флота (командующие эскадрами адмиралы Рожественский и Небогатов) был прорыв во Владивосток. Японский флот (командующий – адмирал Того) имел задачу полного разгрома русского флота. Большая концентрация сил японского флота, его лучшая оснащенность и маневренность привели к военному успеху. Несмотря на мужество и героизм рос-

					<i>ВКР.155518.09.04.04.ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		18

сийских офицеров и матросов, прошедших перед этим 33 тысячи километров от Кронштадта до Цусимы и вступивших в бой с ходу, потери для них были катастрофическими: было потоплено 19 кораблей, 3 крейсера прорвались в нейтральные порты и были интернированы, 2 крейсера и 2 миноносца дошли до Владивостока. Из 14 тысяч человек личного состава эскадр погибло более 5 тысяч [23, 26].

Хроника боя представлена с момента встречи японского флота с русской эскадрой до момента капитуляции нашего командования [9, 14, 45]:

1905.05.27 Японское море. Русская вторая Тихоокеанская эскадра адмирала Рожественского (11 броненосцев, 9 крейсеров, 9 эсминцев, 1 вспомогательный крейсер) встретила с японским флотом адмирала Того (4 броненосца, 24 крейсера, 21 эсминец, 42 миноносца, 24 вспомогательный крейсера) в Цусимском проливе.

7.14. С русской эскадры замечен японский крейсер.

9.40. Обнаружен отряд японских крейсеров.

13.15. Русская эскадра встретила с главными силами японского флота.

13.49. Русские корабли открыли огонь с дистанции 38 кабельтовых.

13.52. Японский флот ответил сосредоточенным огнем по броненосцам «Князь Суворов» и «Ослябя».

14.00. Русскими поврежден и выведен из боя японский крейсер «Асама».

14.25. Получив тяжелые повреждения и потеряв управление, из строя вышел броненосец «Ослябя».

14.30. Выведен из строя и лишился управления броненосец «Князь Суворов».

14.40. Русский броненосец «Ослябя» перевернувшись, затонул.

15.40. Получил тяжелые повреждения эскадренный броненосец «Император Александр III».

16.20. На броненосце «Суворов» из артиллерии уцелело лишь 75-мм орудие в кормовом каземате, которое продолжает стрелять по врагу. Корабль представляет собой сплошной костер от носа до кормы.

					ВКР.155518.09.04.04.ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		19

17.20. Потоплен русский вспомогательный крейсер «Урал».

17.30. Миноносцем «Буйный» сняты с броненосца «Суворов» оставшиеся в живых офицеры штаба и раненый в голову адмирал Рожественский.

18.50. Потоплен броненосец «Император Александр III».

19.00. Раненый вице-адмирал Рожественский передал командование эскадрой контр-адмиралу Небогатову.

19.10. Потоплен броненосец «Бородино».

19.29. Атакован японскими миноносцами и потоплен четырьмя выпущенными в упор торпедами русский броненосец «Князь Суворов»

19.30. Потоплен транспорт «Камчатка».

1905.05.28 Японское море. Ночью японские миноносцы начали атаковать остатки русской второй Тихоокеанской эскадры адмирала Рожественского.

2.15 Был потоплен броненосец «Наварин», русские потопили 3 японских миноносца и повредили 12.

5.00. Южнее Цусима своей командой затоплен русский эсминец «Блестящий».

5.23. Японским крейсером потоплен русский миноносец «Безупречный».

8.00. Севернее Цусима затоплен броненосец «Адмирал Нахимов».

10.05. Японской торпедой потоплен броненосец «Сисой Великий».

10.38. Отряд кораблей адмирала Небогатова (броненосцы «Император Николай I», «Орел», «Генерал-адмирал Апраксин», «Адмирал Сенявин»), окруженный японской эскадрой, капитулировал. Прорваться из японского окружения удалось лишь крейсеру «Изумруд».

11.00. После боя затоплен экипажем крейсер «Светлана».

11.30. Затоплен миноносец «Буйный».

11.50. Затоплен миноносец «Быстрый».

12.43. У побережья Кореи встреченный 3 японскими эсминцами, затоплен своей командой миноносец «Громкий».

14.00. Командой затоплен броненосец «Владимир Мономах».

17.05. На миноносце «Бедовый» сдался в японский плен командующий

					ВКР.155518.09.04.04.ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		20

русской эскадрой вице- адмирал Рожественский.

18.10. Японскими крейсерами «Якумо» и «Ивате» потоплен русский броненосец «Адмирал Ушаков». Из всей второй Тихоокеанской эскадры удалось спастись только нескольким кораблям. Крейсера «Аврора», «Олег» и «Жемчуг» прорвались в г. Манила (Филиппины; США), миноносец «Бодрый», транспорты «Свирь» и «Корея» в г. Шанхай (Китай) где были интернированы, транспорт «Анадырь» ушел к о. Мадагаскар. Во Владивосток прорвались лишь крейсера «Алмаз», «Изумруд» и миноносцы «Бравый» и «Грозный».

Карта боевых действий Цусимского сражения представлена в приложении А.

Построение трехмерных моделей необходимо осуществить с соблюдением точных пропорций и максимальной детализацией реально существовавших образцов. На основании собранных данных о главных силах японского и российского флотов, нами были выбраны основные типы боевых кораблей различных классов [51]. Для российского флота:

- 1) Броненосный крейсер «Аврора»;
- 2) Эскадренный броненосец «Бородино»;
- 3) Броненосец береговой обороны «Адмирал Ушаков»;
- 4) Эскадренный броненосец «Наварин»;
- 5) Эскадренный броненосец «Сисой Великий».

Для японского флота, были выбраны следующие боевые корабли:

- 1) Эскадренный броненосец «Микаса»;
- 2) Броненосный крейсер «Идзумо»;
- 3) Броненосный крейсер «Касуга»;
- 4) Эскадренный броненосец «Фудзи»;
- 5) Эскадренный броненосец «Сикисима».

Для проектирования моделей были собраны схемы и чертежи каждого боевого корабля [38]. Представленные схемы и чертежи дают обзорное техническое представление о каждом корабле, а фотографии периода русско-японской войны дают представление об его историческом облике.

Рассмотрим детально каждый боевой корабль.

Броненосный крейсер «Аврора» – крейсер первого ранга Балтийского флота типа «Диана». Назван в честь парусного фрегата «Аврора», прославившегося при обороне Петропавловска-Камчатского в годы Крымской войны. «Аврора» под командованием капитана 1 ранга И. В. Сухотина вышла из Кронштадта 25 сентября 1903 года и начала переход в Порт-Артур. Но крейсеру не суждено было дойти до места назначения. Вести о вероломном нападении японского флота на Порт-Артур застали «Аврору» в африканском порту Джибути. Дальнейшее плавание было прервано, и корабль вернулся на Балтику 5 апреля 1904 года, где был включен в состав 2-й Тихоокеанской эскадры, формируемой для боевых действий на Дальневосточном морском театре. При подготовке к этому походу произошла смена командиров. В командование «Авророй» с 11 июля вступил капитан 1 ранга К. Р. Егорьев. Изображения и чертежи броненосного крейсера «Аврора» представлены на рисунках 1 и 2, соответственно.

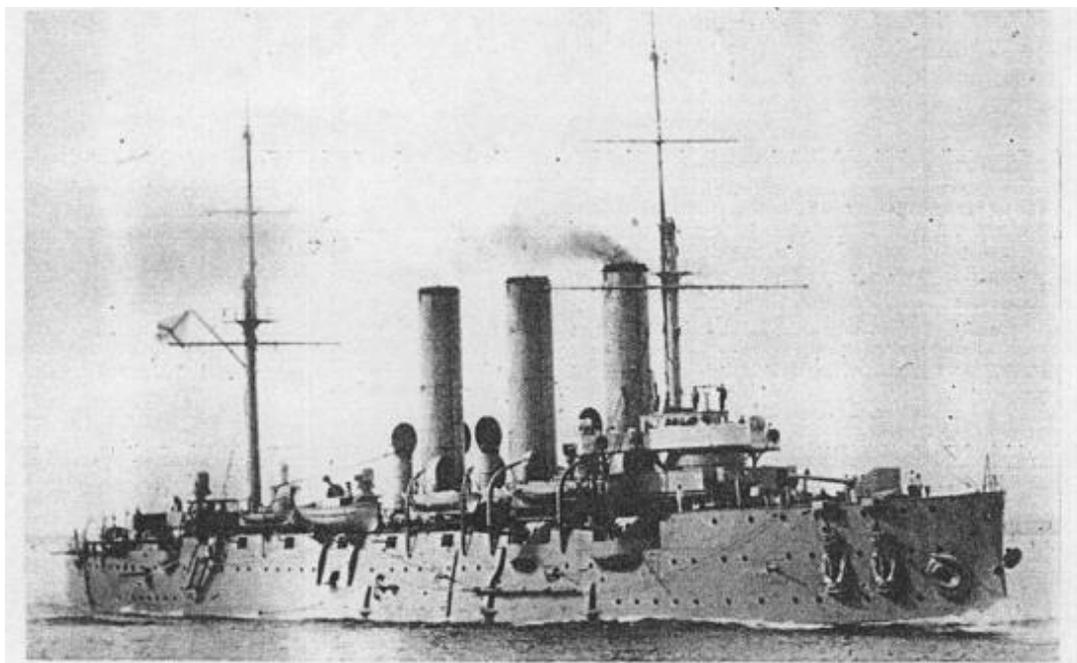


Рисунок 1 – Броненосный крейсер «Аврора»

В Цусимском сражении «Аврора» действовала в составе отряда крейсеров контр-адмирала О. А. Энквиста и постоянно следовала в кильватере за своим

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		22

флагманским кораблем – крейсером «Олег». В ходе Цусимского сражения «Олег» и «Аврора» – бронепалубные, не предназначенные для эскадренного боя крейсера – получили серьезные повреждения и понесли значительные людские потери. В «Аврору» непосредственно попал 21 снаряд, и еще много снарядов разорвалось в воде рядом. По итогам войны крейсер «Аврора» был интернирован. Изображения и чертежи броненосного крейсера «Аврора» представлены на рисунках 1 и 2, соответственно.

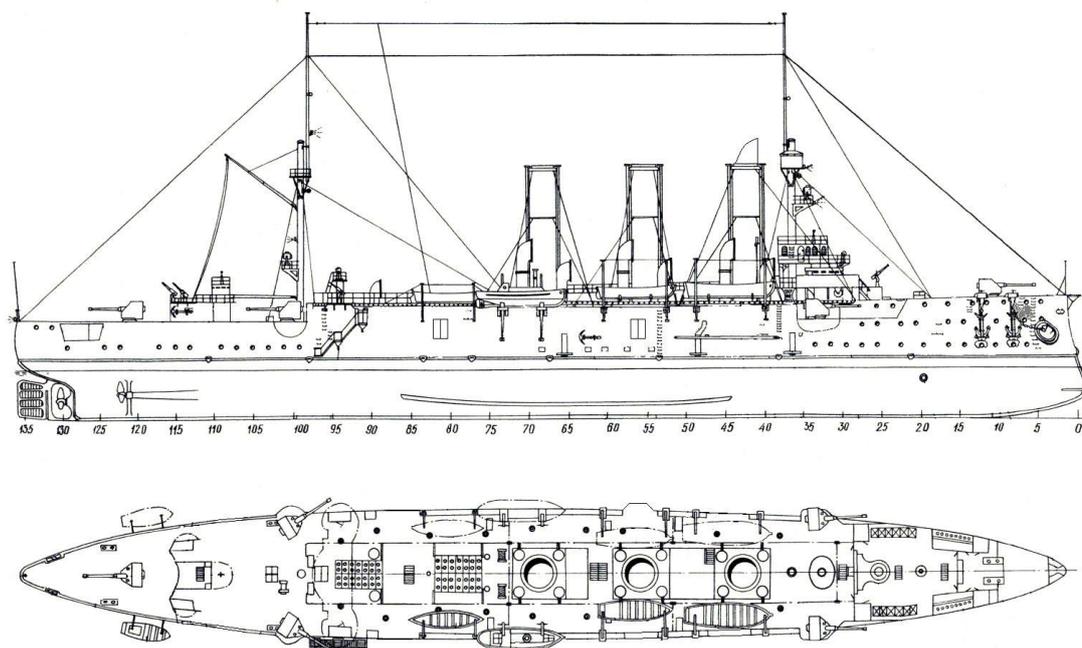


Рисунок 2 – Чертеж броненосного крейсера «Аврора»

Эскадренный броненосец «Бородино» – русский броненосец, построенный в 1901 году в Петербурге на верфи «Новое адмиралтейство» под руководством корабельного инженера Д. В. Скворцова по переработанному проекту броненосца «Цесаревич». Под командованием капитана первого ранга П. И. Серебrenникова участвовал в Цусимском сражении. Из-за ненадёжности машин броненосец мог развивать лишь 16 узлов вместо проектных 18. «Бородино» являлся одним из самых крупных броненосцев, отличаясь наиболее полной системой бронезащиты. Изображения и чертежи эскадренного броненосца «Бородино» представлены на рисунках 3 и 4, соответственно.

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		23

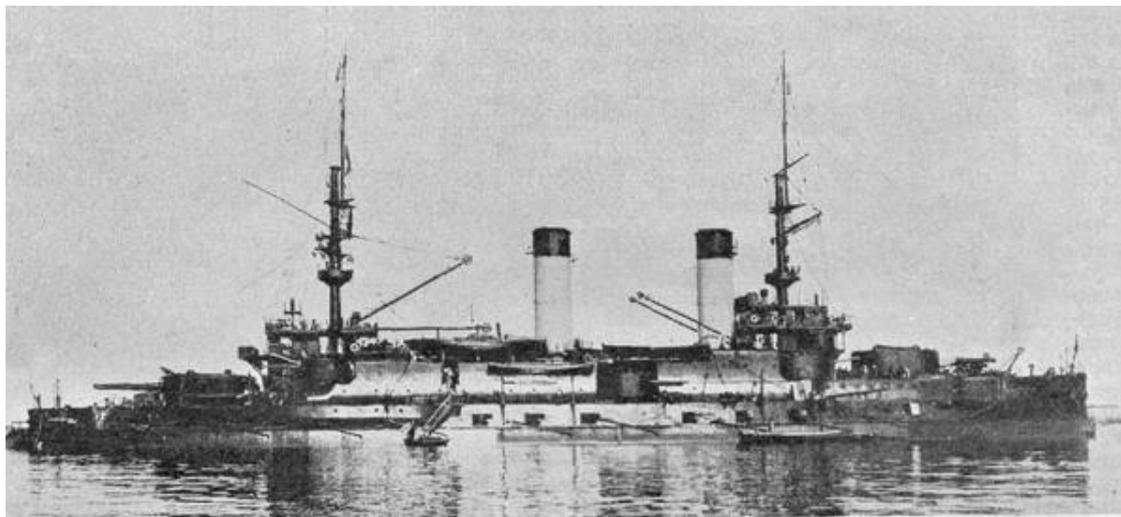


Рисунок 3 – Эскадренный броненосец «Бородино»

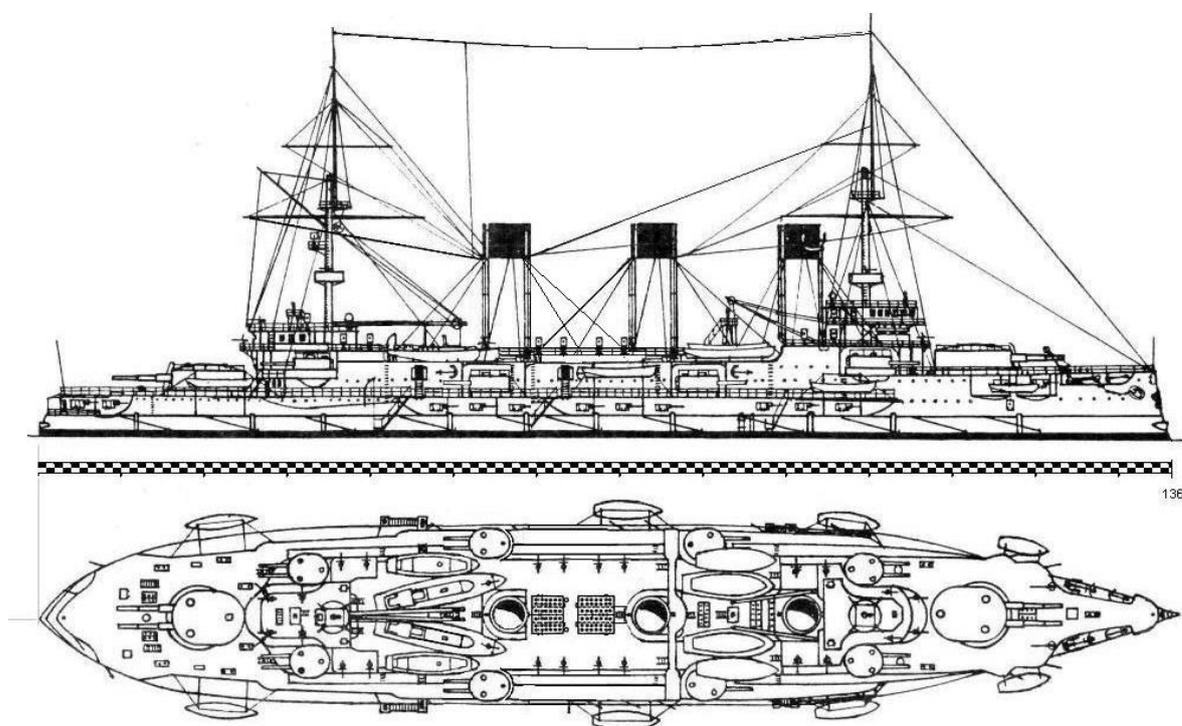


Рисунок 4 – Чертеж эскадренного броненосца «Бородино»

Броненосец береговой обороны «Адмирал Ушаков» – русский броненосец береговой обороны, второй в серии из трёх кораблей («Адмирал Ушаков», «Адмирал Сенявин», «Генерал-адмирал Апраксин»). Участвовал в походе на Дальний Восток в составе кораблей адмирала Н. И. Небогатова, под командованием капитана первого ранга В. Н. Миклухо-Маклая.

В первый день Цусимского сражения оказался под огнём японских кораблей, стрелявших по броненосцу «Александр III». Корабль получил две подвод-

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

24

ные пробоины, вышло из строя одно орудие. Одну пробоину не удалось заделать, в результате чего был затоплен отсек трюма. Скорость упала до 10 узлов, но броненосец остался в строю.

Ночью корабль отстал от основного отряда кораблей и избежал атак японских миноносцев. Утром было решено прорываться к Владивостоку, но по пути корабль встретился с японскими броненосными крейсерами «Ивате» и «Якумо», огневая мощь которых в несколько раз превосходила возможности «Ушакова». В ответ на предложение сдаться командир броненосца приказал открыть огонь. После получасового неравного боя броненосец получил тяжёлые повреждения и был затоплен экипажем. Изображения и чертежи броненосца береговой обороны «Адмирал Ушаков» представлены на рисунках 5 и 6, соответственно.

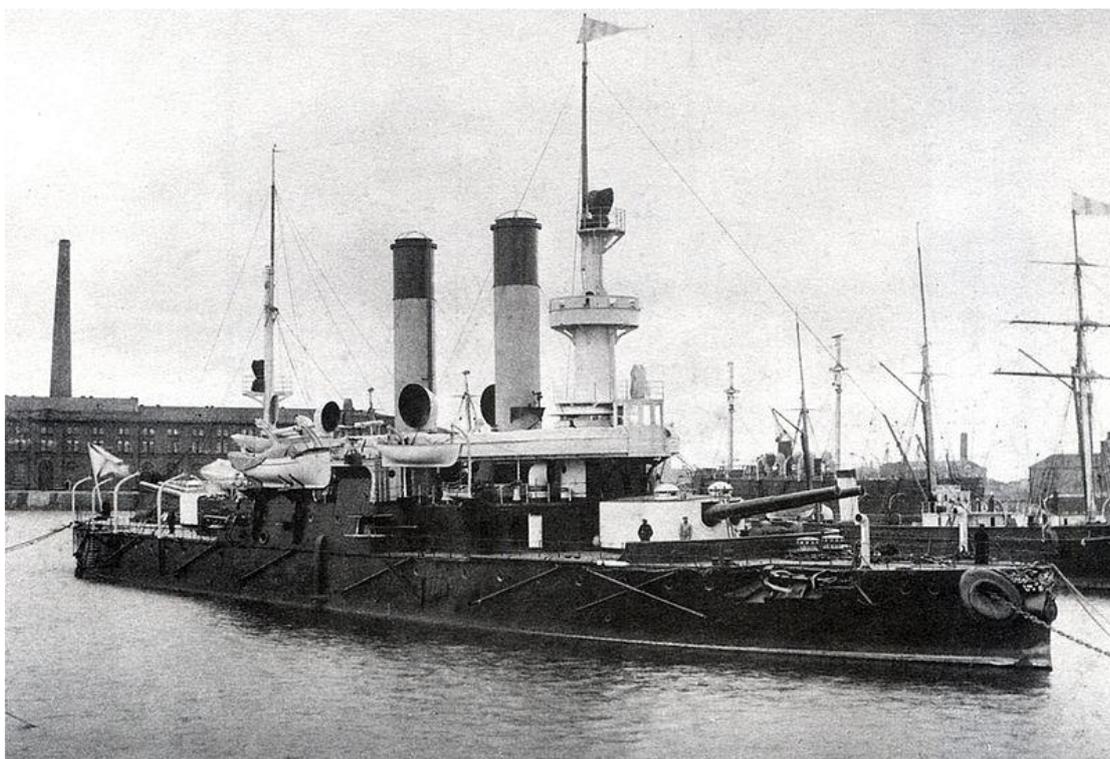


Рисунок 5 – Броненосец береговой обороны «Адмирал Ушаков»

Эскадренный броненосец «Наварин» – первый российский броненосец, имевший классическое вооружение из четырёх тяжёлых орудий главного калибра в двух башнях в носу и корме и батареи пушек среднего калибра в центральной части. Он стал четвёртым броненосцем на Балтийском флоте из по-

строенных по принятой в 1881 году «Двадцатилетней программе», и первым, в проект которого с самого начала закладывалась возможность дальних океанских походов, в том числе и на Дальний Восток.

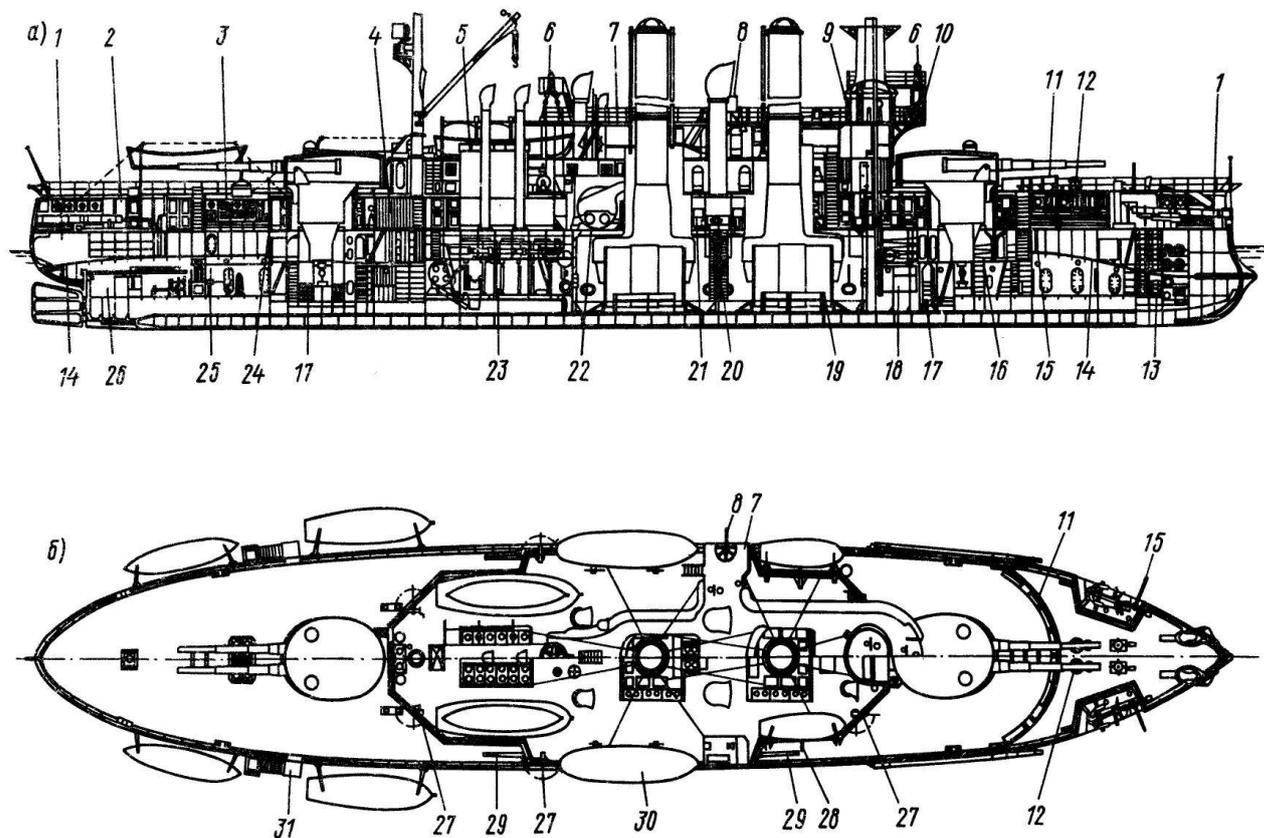


Рисунок 6 – Чертеж броненосца береговой обороны «Адмирал Ушаков»

2 октября 1904 г. вышел из Либавы в составе пятого отряда второй Тихоокеанской эскадры. 14 мая 1905 г. в дневном бою пострадал сравнительно мало. Получил шесть попаданий 8- и 12-дюймовыми снарядами, был смертельно ранен командир корабля капитан 1 ранга барон Б. А. Фитингоф, убиты или ранены еще 17 человек. Ночью был торпедирован в корму с правого борта, а затем еще в середину правого борта и в левый борт с носа. Перевернулся и затонул с 700 членами экипажа. Спасено 3 человека. По другим данным после попадания первой торпеды подорвался на 1 или 2 минах, поставленных по его курсу японскими миноносцами. Изображения и чертежи эскадренного броненосца «Наварин» представлены на рисунках 7 и 8, соответственно.

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

26



Рисунок 7 – Эскадренный броненосец «Наварин»

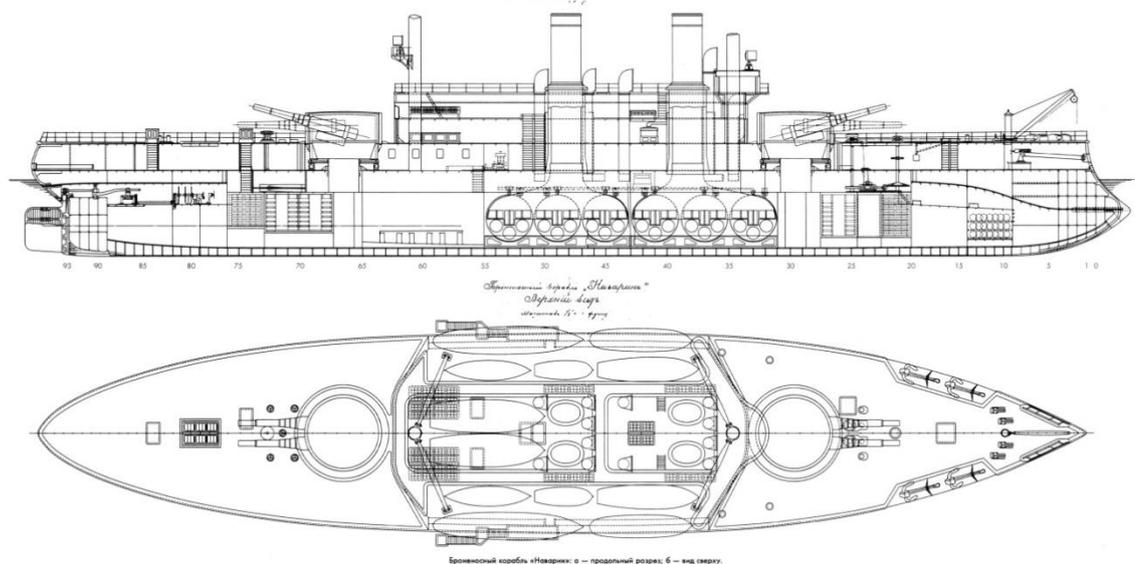


Рисунок 8 – Чертеж эскадренного броненосца «Наварин»

Эскадренный броненосец «Сисой Великий» – пятый из броненосцев Балтийского флота, построенных по принятой в 1881 году двадцатилетней судостроительной программе. В ходе дневного боя корабль, шедший в составе второго броненосного отряда, получил достаточно тяжелые повреждения. Ночью броненосец был атакован японскими миноносцами и получил торпедное попадание оказавшееся для него смертельным. Утром 15 мая с сильным креном корабль достиг острова Цусима, где был встречен в 7 ч 20 мин тремя японскими

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

27

вспомогательными крейсерами и миноносцем. Утром 15 мая 1905 броненосец затонул. Изображения и чертежи эскадренного броненосца «Сисой Великий» представлены на рисунках 9 и 10, соответственно.

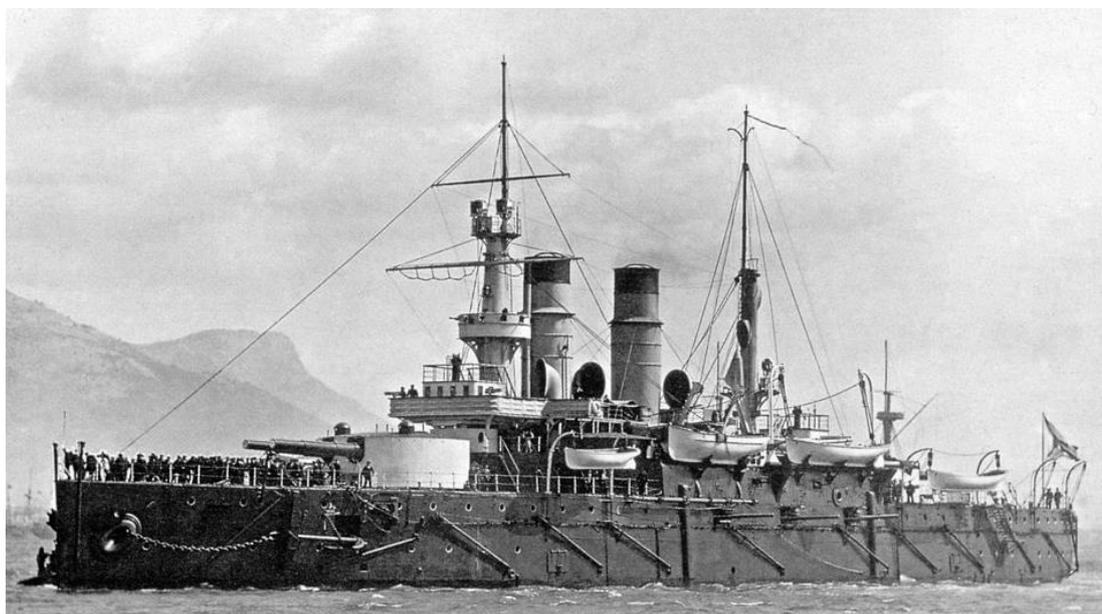


Рисунок 9 – Эскадренный броненосец «Сисой Великий»

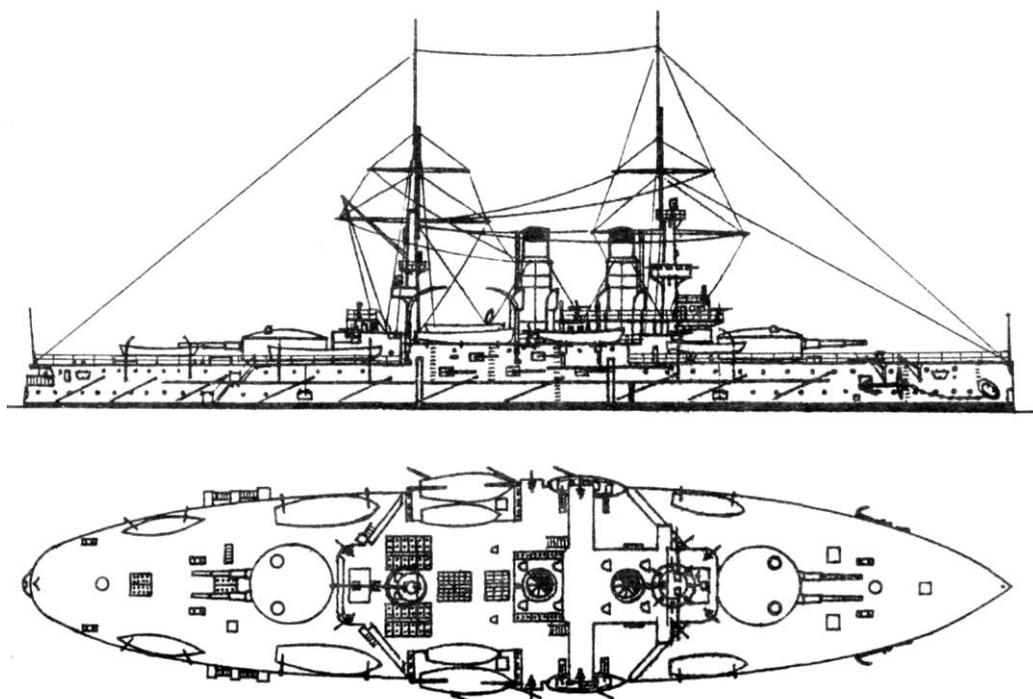


Рисунок 10 – Чертеж эскадренного броненосца «Сисой Великий»

Первым боевым кораблем японского флота был выбран эскадренный броненосец «Микаса» – флагман японского флота. Участвовал в Цусимском сра-

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

28

жени. Корабль затонул в результате пожара и взрыва пороховых погребов вскоре после окончания войны (11 сентября 1905 года). После нескольких попыток в августе 1906 года был поднят на поверхность и после двухлетнего ремонта возвратился на службу. В 1923 корабль был выведен из состава флота и превращён в корабль-музей. Изображения и чертежи эскадренного броненосца «Микаса» представлены на рисунках 11 и 12, соответственно.

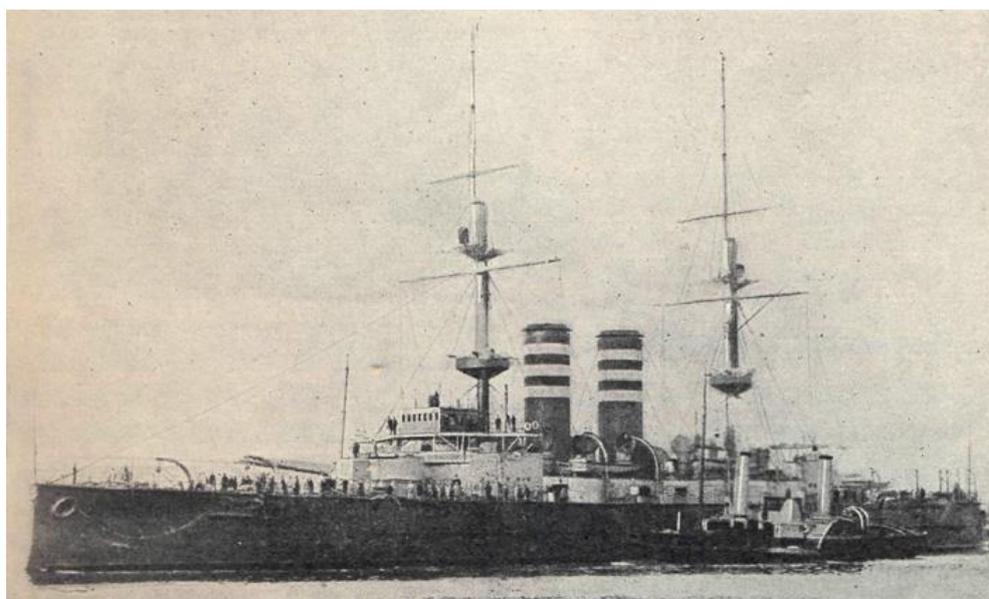


Рисунок 11 – Эскадренный броненосец «Микаса»

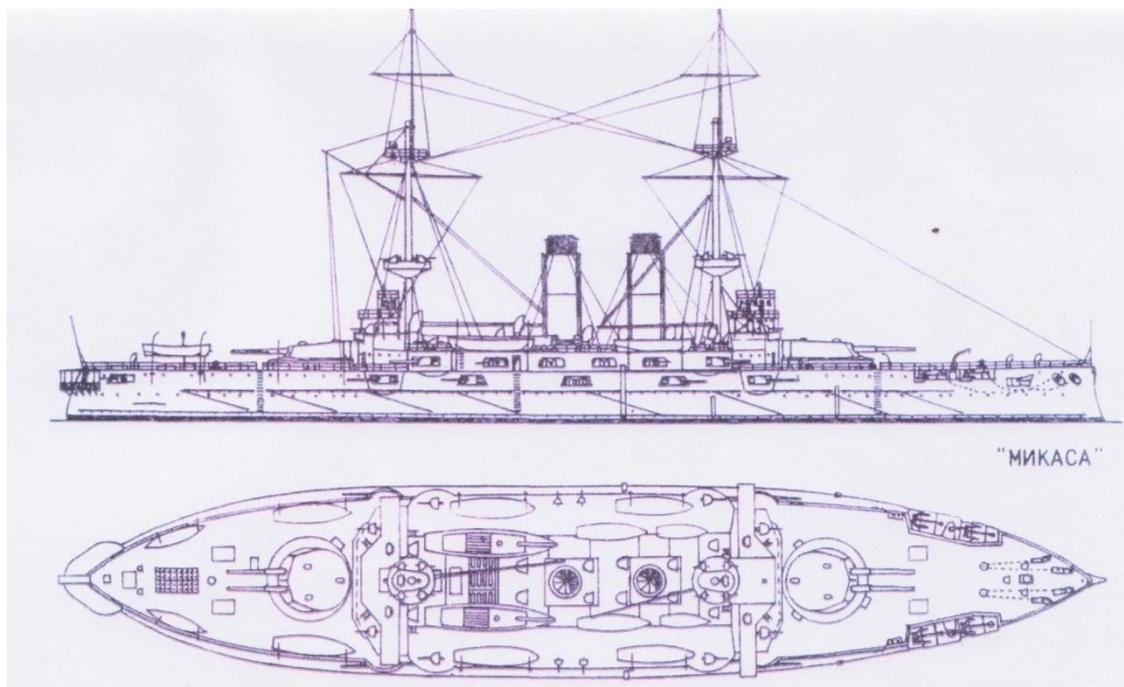


Рисунок 12 – Чертеж эскадренного броненосца «Микаса»

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

29

Броненосный крейсер «Идзумо» – японский броненосный крейсер, участвовавший в Русско-японской войне. Был назван в честь исторической провинции Идзумо. Во время Русско-японской войны был флагманским кораблем вице-адмирала Камимурэ Хиконодзэ. Был поврежден в бою 14 августа 1904 г. – 20 попаданий (2 убито, 17 ранено). В Цусимском бою получил 12 снарядов, убито 3 и ранено 27 человек. Изображения и чертежи броненосного крейсера «Идзумо» представлены на рисунках 13 и 14, соответственно.

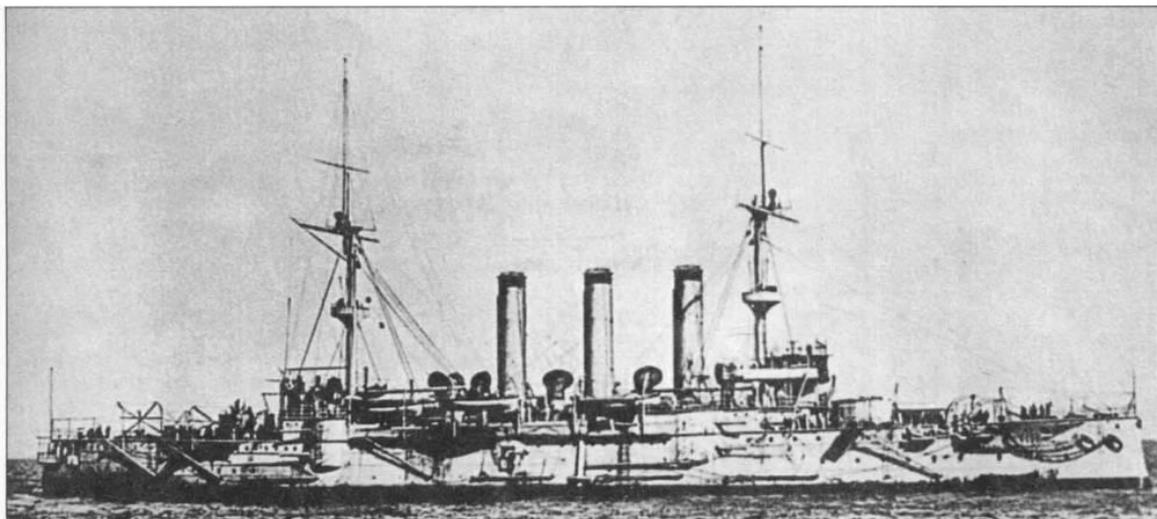


Рисунок 13 – Броненосный крейсер «Идзумо»

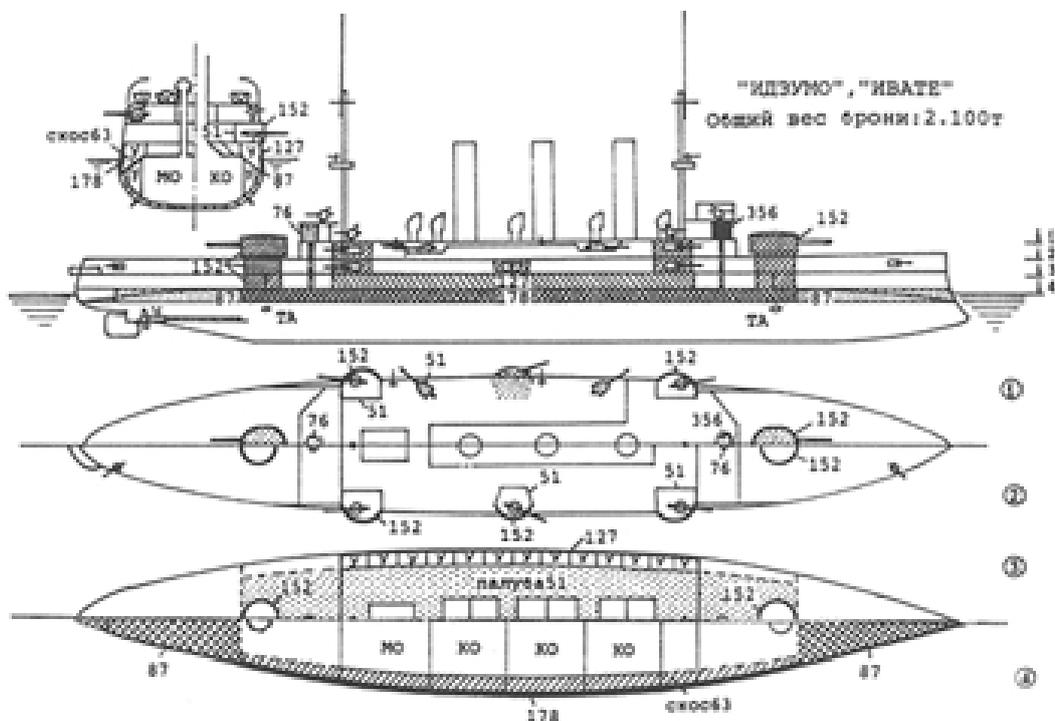


Рисунок 14 – Чертеж броненосного крейсера «Идзумо»

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

30

Броненосный крейсер «Касуга» – японский броненосный крейсер, участник русско-японской войны. Головной корабль своего типа. От других японских кораблей этого класса отличался более тонким бронированием, но по большей площади. Скорость на службе не превышала 18 – 18,5 узлов. 14 мая 1905 г. броненосный крейсер участвовал в Цусимском сражении, получил повреждения (3 попадания (7 убитых, 20 раненых)). Изображения и чертежи броненосного крейсера «Касуга» представлены на рисунках 15 и 16, соответственно.

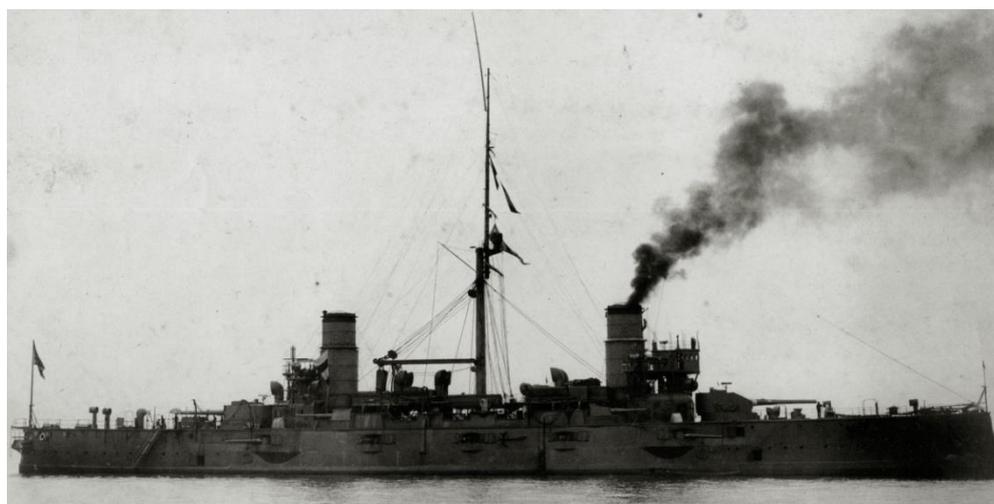


Рисунок 15 – Броненосный крейсер «Касуга»

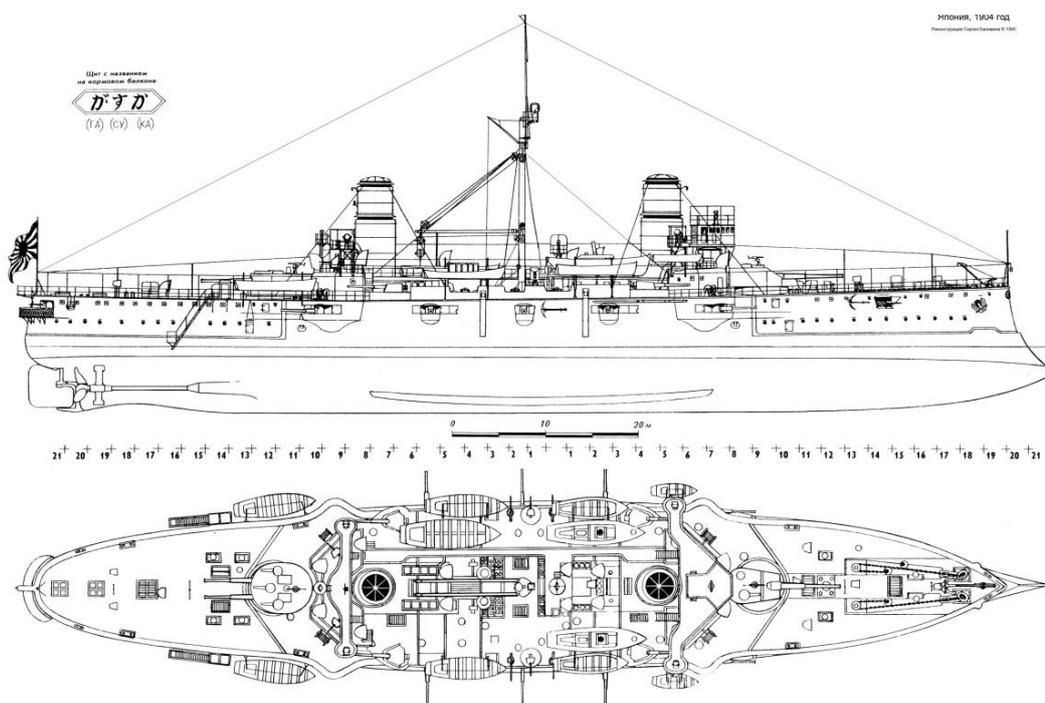


Рисунок 16 – Чертеж броненосного крейсера «Касуга»

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

31

Эскадренный броненосец «Фудзи» – броненосец японского флота, названный в честь вулкана Фудзи на острове Хонсю. Во время русско-японской войны «Фудзи» под командованием капитана 1 ранга К. Мацумото входил в первый боевой отряд первой эскадры, участвовал в обстрелах и блокаде Порт-Артура. В Цусимском сражении после попадания с «Фудзи» взорвался и затонул русский броненосец «Бородино». Сам Фудзи в бою получил 11 попаданий, но корабль чудом избежал гибели после взрыва в кормовой башне. Изображения и чертежи эскадренного броненосца «Фудзи» представлены на рисунках 17 и 18, соответственно.

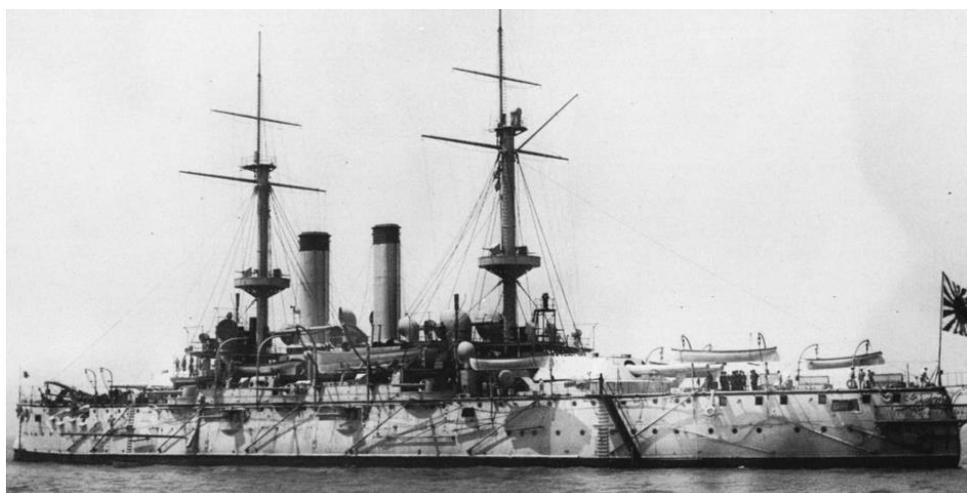


Рисунок 17 – Эскадренный броненосец «Фудзи»

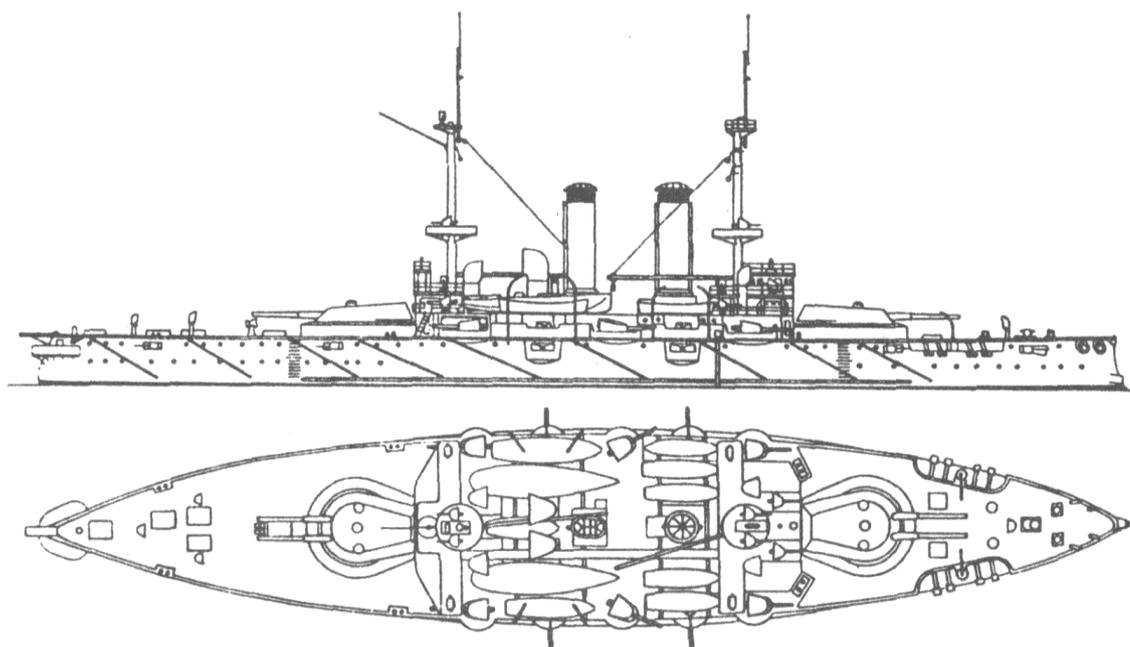


Рисунок 18 – Чертеж эскадренного броненосца «Фудзи»

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

32

Эскадренный броненосец «Сикисима» – японский эскадренный броненосец, головной в серии из двух единиц (второй – «Хацусэ»). Участвовал в Русско-японской войне. Во время Цусимского сражения был вторым мателотом после флагманского броненосца «Микаса» и подвергался интенсивному артиллерийскому обстрелу. В результате корабль получил 11 попаданий, самым тяжелым из которых было попадание непосредственно под амбразуру одного из 6" орудий, выведшее из строя всю прислугу этого орудия. Потери среди команды составили 13 убитых, 24 раненых. Изображения и чертежи эскадренного броненосца «Сикисима» представлены на рисунках 19 и 20, соответственно.

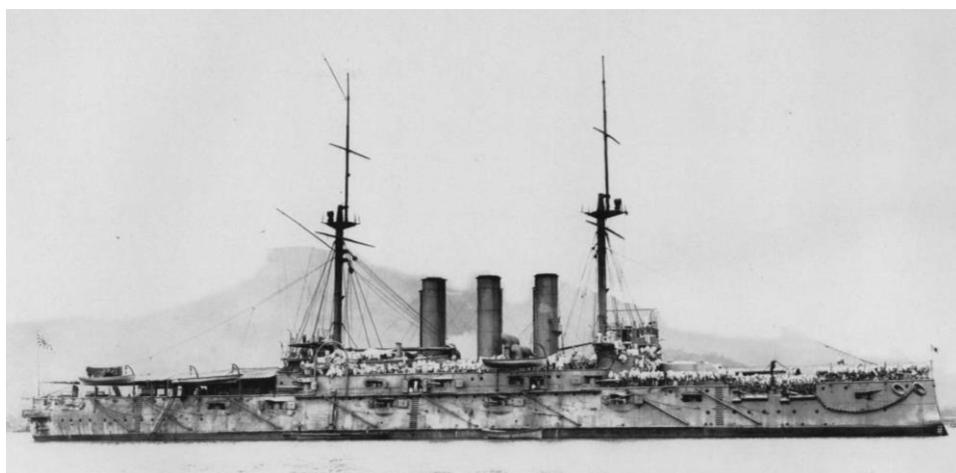


Рисунок 19 – Эскадренный броненосец «Сикисима»

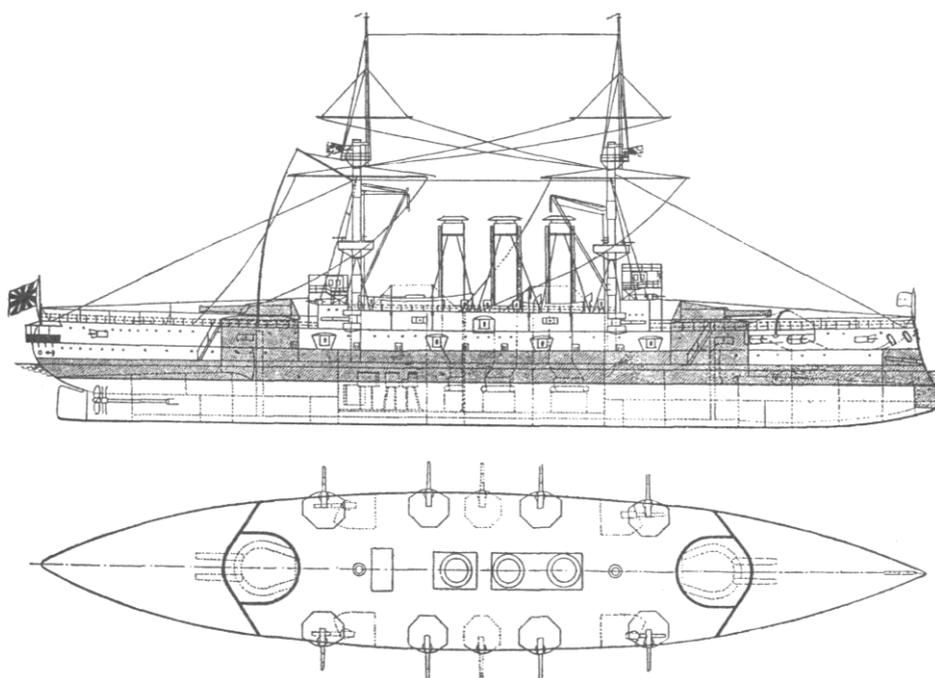


Рисунок 20 – Чертеж эскадренного броненосца «Сикисима»

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

33

Собранные чертежи и фотографии боевых кораблей позволят восстановить внешний облик морских судов, а также послужат основой при построении реалистичных 3D-моделей.

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		34

2 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РЕШЕНИЯ ЗАДАЧИ

Прикладное программное обеспечение используется для решения задач в различных областях применения компьютерных систем обработки данных. Прикладная программа или приложение – это программа, предназначенная для решения задачи или класса задач и проблемной области – то есть конкретной области применения информационных технологий. [37]

Для создания 3D-моделей костей нами был выбран пакет для создания трёхмерной компьютерной графики Blender, включающий в себя средства моделирования, анимации и рендеринга. Дальнейшая разработка информационной системы осуществлялась посредством использования мультиплатформенного графического процессора Unity3D с применением высокоуровневого языка программирования C#. Для хранения данных, используемых для трёхмерной визуализации, использовалась реляционная база данных SqliteDog.

Рассмотрим подробнее комплекс программ необходимых для реализации программного модуля.

2.1 Обоснование выбора программы для 3D моделирования

Для создания сложных полигональных 3D-моделей служат специализированные программные пакеты трёхмерного моделирования. При выборе универсального 3D-редактора учитываются различные факторы: цель моделирования (что конкретно необходимо сделать); функциональные возможности; удобство использования и т.д.. С целью выбора подходящего программного пакета трёхмерной графики необходимо провести анализ популярных в настоящее время 3D-редакторов: 3Ds Max, Cinema 4D и Blender. В таблице 1 представлены сравнительные характеристики трёхмерных редакторов.

Таблица 1 – Основные характеристики пакетов трёхмерной графики

Характеристики	Наименование 3D-редактора		
	3Ds Max	Cinema 4D	Blender
1	2	3	4
1 Моделирование	+	+	+

1	2	3	4
2 Скульптинг	-	-	+
3 Текстурирование	+	+	+
4 Инструменты анимации	+	+	+
5 Динамика твердых и мягких тел	+	+	+
6 Система частиц	+	+	+
7 Поддержка скриптов	+	+	+
8 Поддерживаемые операционные системы	Windows	Windows, Mac OS	Windows, Mac OS, Linux
9 Наличие бесплатной версии	Пробная – 30 дней, Студенческая – 3 года (некоммерческое использование)	Пробная – 30 дней, Студенческая – 18 месяцев (некоммерческое использование)	В свободном доступе
10 Наличие документации в базовой поставке	+	+	-
11 Размер программы	Относительно большой	Относительно большой	Небольшой

Исследование программных пакетов трехмерного моделирования показало, что наиболее подходящим по своим функциональным возможностям является свободно распространяемый пакет для создания трёхмерной компьютерной графики Blender, мощный и удобный инструмент для моделирования, обработки видео и звука, анимации, рендеринга. Также Blender весит всего на всего несколько десятков мегабайт (за счет отсутствия громоздких систем защиты и регистрации, а так же предустановленных сцен) и переведен на 26 языков, в том числе и на русский. Несомненно, это обеспечивает удобство в работе и делает ее более плодотворной, поскольку не возникает трудностей в плане языкового барьера. Таким образом, нам становится доступным бесплатный

кроссплатформенный пакет для трехмерного моделирования профессионального уровня.

Пакет Blender был разработан компанией NeoGeo, успешным создателем трехмерных анимаций. На данный момент программа является наиболее популярной среди бесплатных и свободно распространяемых 3D редакторов. Быстрый и стабильный рост прикладного пакета обеспечивает профессиональная команда разработчиков. Blender содержит самые современные средства для художников и специалистов в области мультимедиа.

Характерной особенностью пакета Blender является его кроссплатформенность, имеется поддержка различных операционных систем, таких как: Windows, Linux, Mac OS. Также неоспоримым плюсом является небольшой размер исполняемого файла программы (10 – 15 Мбайт) по сравнению с другими популярными пакетами для 3D моделирования.

Пакет Blender поддерживает различные геометрические примитивы, а также полигональные модели, систему быстрого моделирования (SubSurf), кривые Безье, поверхности NURBS, метасферы, скульптурное моделирование и векторные шрифты, содержит универсальные встроенные механизмы рендеринга и интеграция с внешним рендерером YafRay, LuxRender [27].

Возможности создания анимации пакета включают в себя: инверсную кинематику, скелетную анимацию и сеточную деформацию, анимацию по ключевым кадрам и нелинейную анимацию. Также имеются возможность редактирования весовых коэффициентов вершин, система волос на основе частиц, динамика мягких и твердых тел. Есть возможности управления скелетной деформацией, создания быстрой анимации двуногих созданий, управления физическими силами, действующими на персонажей.

С выходом обновлений пакета, постоянно улучшаются функциональные возможности программы, добавляются различные демонстрационные и эффекты, расширяются типовые библиотеки. Встроенные функции полигонального моделирования, операции по созданию трудоемких объектов, средства двумерного моделирования, различные модификаторы для работы с геометрией моде-

ли, обширные возможности творческой работы с текстурами – все это делает работу в Blender более эффективной и удобной.

Также неоспоримым достоинством данного пакета для 3D моделирования является элегантный внешний вид, удобный и дружелюбный интерфейс – на рабочую панель вынесены минимальное количество требуемых при работе интуитивно ясных клавиш, для работы с которыми можно воспользоваться как обыкновенной для нас мышью, но и графическим планшетом. Интерфейс программы Blender представлен на рисунке 21.

В программе предусмотрена внутренняя файловая система, позволяющая хранить несколько сцен в едином файле (называемом .blend файл). Все «.blend» файлы совместимы как с более старыми, так и с более новыми версиями Blender. Так же все они переносимы с одной платформы на другую и могут использоваться как средство переноса созданных ранее работ.

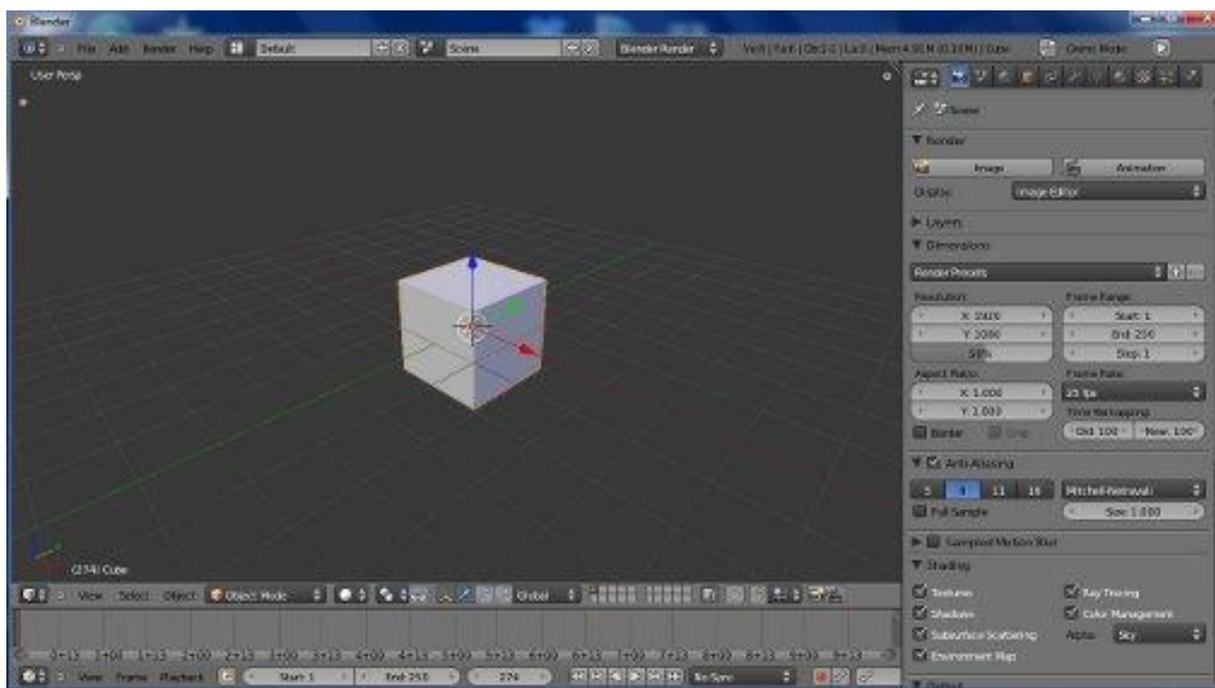


Рисунок 21 – Интерфейс программы Blender

Blender делает резервные копии проектов во время всей работы программы, что позволяет сохранить данные при непредвиденных обстоятельствах. Все сцены, объекты, материалы, текстуры, звуки, изображения, эффекты могут быть сохранены в единый «.blend» файл., что очень удобно. Настройки рабочей

среды могут быть сохранены в «.blend» файл, благодаря чему при загрузке файла вы получите именно то, что сохранили в него. Файл можно сохранить как «пользовательский по умолчанию», и каждый раз при запуске Blender вы будете получать необходимый набор объектов и подготовленный к работе интерфейс.

Blender – это компетентный пакет для полноценной работы с 3d-графикой, имеющий сильный инструментарий. Система рендеринга в пакете Blender имеет три встроенных обработчика, которые находятся в меню Engine (Движок), расположенном в заголовке главного окна программы (рисунок 22).

Blender Render – это самый первый визуализатор программы, используемый по умолчанию. Движком для визуализации сцены в режиме реального времени, благодаря которому можно создать игру или приложение, выступает Blender Game. Cycles Render – это новый рендер, появившийся в версии программы 2.61.[29].

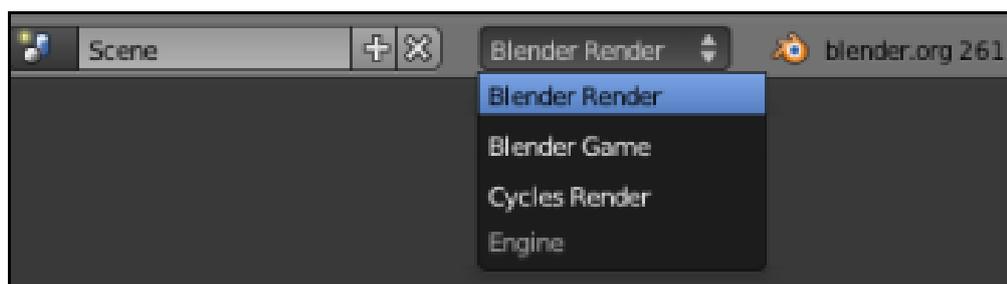


Рисунок 22 – Встроенные рендеры Blender

Помимо встроенных обработчиков, в программе можно использовать другие популярными рендеринги: YafRay, LuxRender, Pov-Ray, Rendermfarm.

Эти движки подключаются к программе с помощью встроенного механизма плагинов. Также Blender поддерживает технологию смазывания движения (Motion Blur), позволяющее отобразить движение в кадре без рывков. Технически это выглядит, как создание дополнительных шагов движения объектов в одном и том же кадре.

Можно с полной уверенностью заявить, что последние версии программы Blender содержат безусловно все, требуемые для работы, модификаторы. Это категории модификаторов выбора, сеток, полигонов, оптимизации плоскости и

прочие. А если учитывать, что использование любого модификатора предполагает установку некоего количества пользовательских характеристик, становится понятно, что работа в Blender сравнима с творчеством и раскрывает перед пользователем безграничное число возможностей для реализации его задумок.

2.2 Обоснование выбора визуализатора

От правильного выбора инструмента для создания качественного пользовательского интерфейса зависит судьба проекта разработки. Данный выбор зависит от многих факторов.

По результатам работы планируется создание полноценного интерактивного программного продукта для персонального компьютера. Необходимо выбрать графический процессор, который будет соответствовать всем предъявляемым требованиям. С целью выбора подходящего визуализатора необходимо провести анализ популярных в настоящее время графических процессоров: Unity3D, Unreal Engine и CryEngine. В таблице 2 представлены сравнительные характеристики графических процессоров.

Таблица 2 – Основные характеристики графических процессоров

Характеристики	Наименование графических процессоров		
	Unity3D	Unreal Engine	CryEngine
1	2	3	4
1 Язык программирования	C#,JavaScript, Boo	C++	Lua Script
2 Физический движок	PhysX	PhysX	Soft-body
3 Импорт из Blender	+	+	+
4 Наличие бесплатной лицензии	+	+	+
5 Поддерживаемые операционные системы	Windows, Mac OS, Linux	Windows, Mac OS, Linux	Windows, Linux
6 Графика	DirectX и OpenGL	DirectX и OpenGL	DirectX и OpenGL
7 Системы частиц	+	+	+

1	2	3	4
8 Моделирование	2D, 3D	2D, 3D	3D
9 Система рендеринга	OpenGL 2.1, OpenGL ES, DirectX 9.0c ++, Deferred Rendering, Multithreaded rendering, HDR rendering и т.д	Direct3D, OpenGL, Pico-matic; Glide, S3, PowerVR	OpenGL и DirectX 8/9, Xbox GameCube, Xbox 360.
10 Порог вхождения	Низкий	Средний	Высокий

По результату сравнительного анализа выявлено, что наиболее подходящей по своим функциональным возможностям средой для визуализации пользовательского интерфейса является свободно распространяемый графический процессор Unity3D. Основанием выбора послужили: мультиплатформенность, простой настраиваемый интерфейс, большое количество обучающего материала и легкость в освоении.

Кроссплатформенный графический процессор Unity3D имеет простой в обращении интерфейс, позволяющий с легкостью проводить настройку по желанию пользователя; меню представляет собой совокупность рабочих окон, которые могут быть дополнены либо заменены другими. Все это позволяет производить отладку приложения прямо в самом редакторе.

В настоящий момент графической процессор позволяет создавать программы, которые могут выполняться под управлением ОС Microsoft Windows, Linux, Apple iOS и Android. Приложения, созданные в среде Unity3D, поддерживают рендеринг с помощью интерфейсов программирования приложений на основе 11-ой версии API DirectX и OpenGL [52]. Также Unity3D поддерживает интеграцию с системой моделирования физических свойств объектов – PhysX, физику твёрдых тел и ткани, и физику типа Ragdoll («тряпичная кукла»). В редакторе процессора содержится система наследования объектов; благодаря ко-

торой дочерние объекты повторяют все изменения родительского объекта (позиции, поворот и масштаб).

Unity3D разрешает использовать основные форматы хранения информации о трехмерных объектах, в частности: *.fbx, *.3ds, и *.dae.

Кроме типовых для Unity3D встроенных программных библиотек в состав системы входит редактор сцены «WYSIWYG», позволяющий располагать, отражать и задавать свойства объектов прямо в процессе разработки [53]. Редактор сцены представлен на рисунке 23.

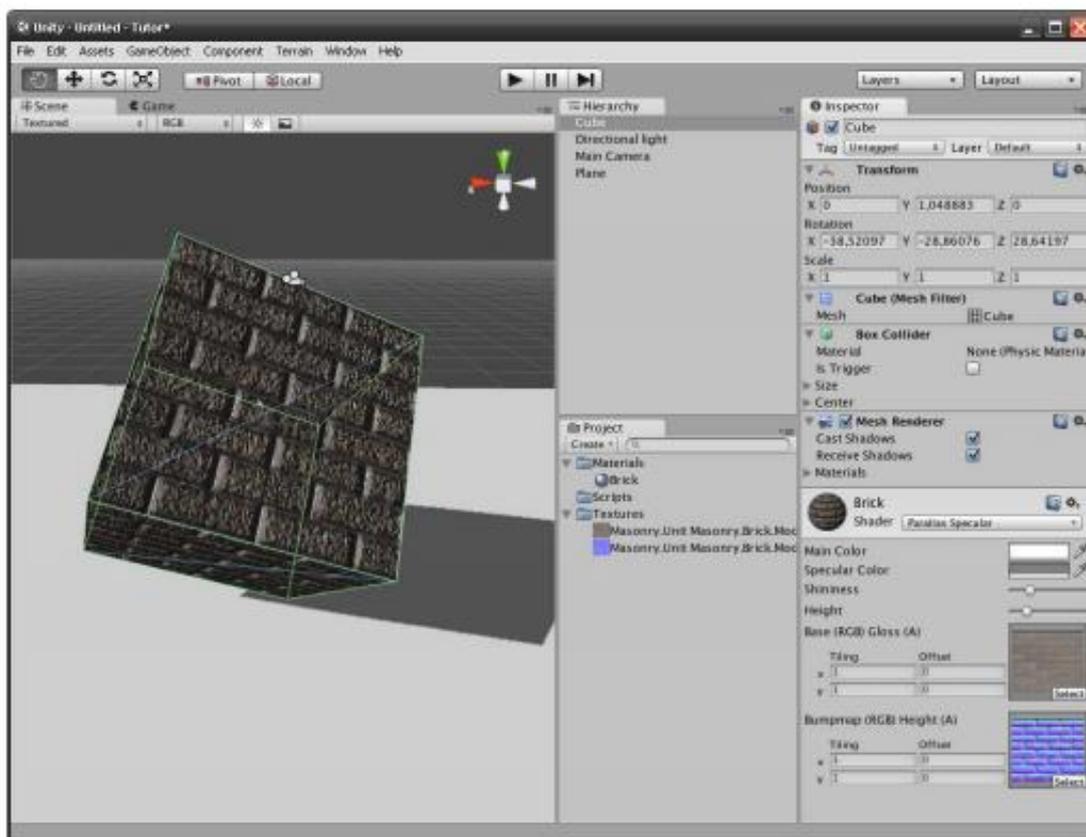


Рисунок 23 – Редактор сцены Unity3D

Вдобавок вместе с редактором Unity3D прилагается интегрированная среда разработки «MonoDevelop» (совместима с .NET Framework 2.0), предназначенная для расширения характеристик как встроенных в систему объектов, так и объектов, создаваемых пользователем. Помимо этого имеется функция пошагового выполнения, т.е. когда выполнение приостановится, становится доступной шкала отладочных инструментов. Это довольно удобный способ для осу-

Изм.	Лист	№ докум.	Подп.	Дата

ществления отладки исходного кода. Редактор кода MonoDevelop представлен на рисунке 24.

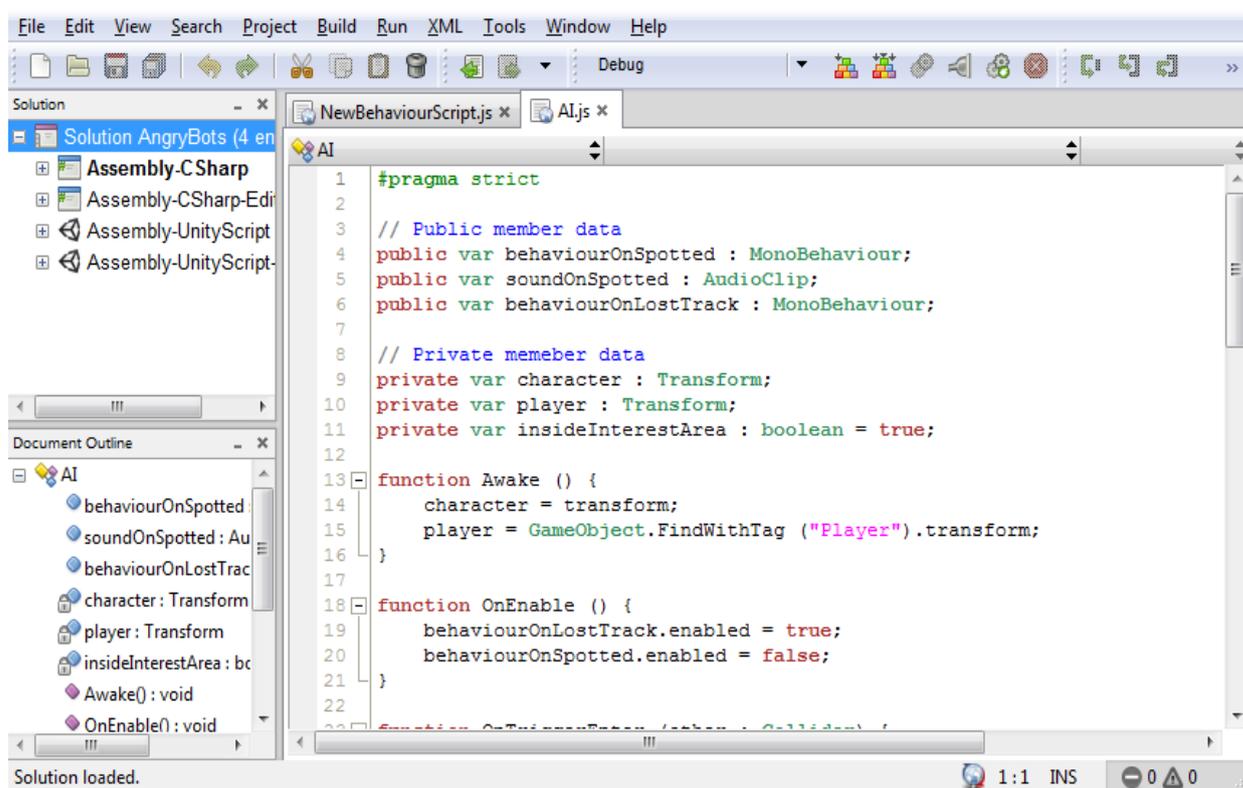


Рисунок 24 – Редактор MonoDevelop

Стандартным языком программирования для графического процессора Unity3D является объектно-ориентированный язык C#. Данный язык основан на строгой компонентной архитектуре и реализует передовые механизмы обеспечения безопасности кода. Язык C# имеет отличную поддержку компонентов, надежен и устойчив благодаря использованию «сборки мусора», обработки исключений, безопасности типов [39].

Язык C# разрабатывался «с нуля» и вообрал в себя много полезных свойств таких языков, как C++, Java, Visual Basic, Pascal и др. Выделение и объединение лучших идей современных языков программирования делает язык C# языком программирования нового поколения.

Проект в Unity3D подразделяется на уровни (сцены), специальные файлы с набором объектов, сценариев, и настроек. Сцены могут включать в себя полноценные объекты (модели) и пустые объекты (не имеющие моделей). Объекты

также содержат наборы компонентов для взаимодействия скриптов. Объекты имеют свое название (допускается наличие двух и более объектов с одинаковыми названиями), собственную метку и слой для отображения. Так, у любого объекта на сцене присутствует компонент Transform, хранящий координаты местоположения, поворота и размеров объекта по трём осям. У объектов с видимой геометрией также по умолчанию присутствует компонент Mesh Renderer, делающий модель объекта видимой.

Среда разработки Unity3D позволяет разрабатывать двух и трехмерные кроссплатформенные приложения. Возможность переноса программного продукта на множество современных платформ является одним из важнейших достоинств данного программного продукта.

Подводя итог, хочется отметить, что среда разработки Unity3D является наиболее подходящей средой для визуализации пользовательского интерфейса, в которой приятно сочетаются приятный пользовательский интерфейс, понятные панели инструментов и достаточно широкий выбор языков для программирования сценариев.

2.3 Обоснование выбора СУБД

В качестве источника данных необходимо выбрать наиболее подходящую реляционную базу данных, с возможностью взаимодействия с графическим процессором Unity3D. Проведя анализ ныне существующих на рынке реляционных баз данных, были выделены основные базы данных SQLite и MySQL. В таблице 3 представлены сравнительные характеристики реляционных баз данных.

Таблица 3 – Основные характеристики реляционных баз данных

Характеристики	Наименование реляционных баз данных	
	MySQL	SQLite
1	2	3
1 Архитектура	Клиент-сервер	Встраиваемая БД

1	2	3
2 Документация	Англоязычная документация	Англоязычная документация
3 Поддерживаемые операционные системы	Windows, Mac OS, Linux	Windows, Mac OS, Linux
4 Простота разработки и администрирования	Администрирование через консоль, множество приложений для разных платформ для локального администрирования, администрирование через веб-сервер	Администрирование через консоль, множество приложений для разных платформ для локального администрирования, администрирование через веб-сервер.
5 Надёжность хранения данных	Надёжность обеспечивается сервером баз данных. Имеются механизмы резервного копирования	Вся база хранится в одном файле.
6 Визуальные инструменты создания и редактирования объектов базы данных	+	+
7 Возможность ввода и редактирования данных в табличном режиме	+	+
8 Русскоязычный интерфейс	+	+
9 Доступность	Бесплатное распространение, есть коммерческая версия	Бесплатное распространение

Анализ функциональных возможностей рассматриваемых реляционных баз данных позволил выделить наиболее подходящую базу данных SQLite, ввиду её высокой популярности, относительно высокой скорости работы, наличию большого количества документации и удобного интерфейса.

SQLite – это кроссплатформенная встраиваемая реляционная база данных, исходный код которой, написанный на языке Си, находится в свободном доступе [25].

SQLite представляет собой библиотеку, работающую в том же процессе, что и само приложение [17]. Это существенно упрощает работу программы и уменьшает время ее отклика. Имеется возможность хранения всех данных в одном файле на компьютере, благодаря чему этот файл можно с легкостью перенести на другой персональный компьютер.

Её признанием на популярность считается её удобный интерфейс, возможность работы без дополнительной установки и достаточно хороший показатель поиска и записи среди встраиваемых баз данных. Интерфейс встраиваемой реляционной базы данных SQLite представлен на рисунке 25.

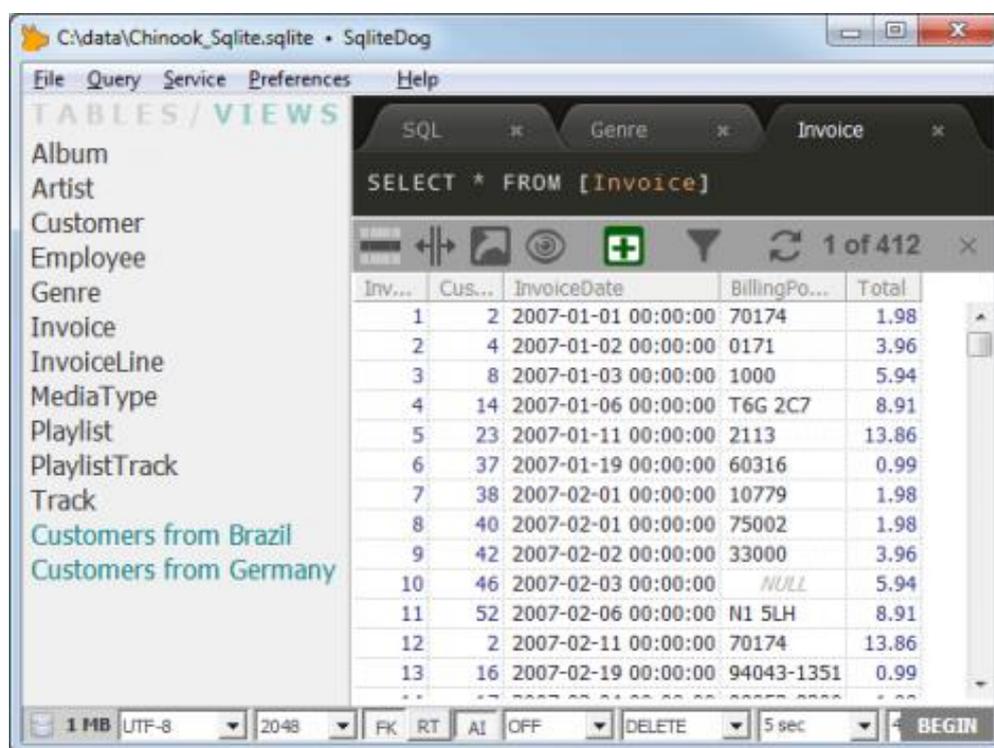


Рисунок 25 – Интерфейс SQLite

Используя высоко эффективную инфраструктуру, SQLite имеет возможность действовать в крошечном объёме выделяемой для неё памяти, во много раз наименьшем, нежели в каких-либо иных системах БД [19]. Это делает SQLite очень удобным инструментом с возможностью применения практически в любых задачах, возлагаемых на базу данных.

Существуют некоторые особенности при работе SQLite. Следует помнить, что чтение из одной базы данных осуществляется несколькими процессами, но

запись производится только в том случае, если в данный момент не выполняются другие запросы.

Функциональной клиентской частью реляционной базы данных является исполняемый файл `sqlite3`. Это консольная утилита, позволяющая работать с файлом базы данных непосредственно из командной строки [44]. Вместе с тем данные в СУБД могут быть типизированы динамически. Получается, что можно вносить информацию в столбец таблицы в не зависимости от его типа.

Помимо всего прочего SQLite отличается быстротой, а также мощностью, благодаря технологиям обслуживающих библиотек. Также среди достоинств SQLite можно отметить: отсутствие необходимости настройки сервера; полностью свободная лицензия; поддержка транзакций, триггеров, представлений, вложенных запросов и возможность использования различных языков программирования.

Выбор именно этой СУБД в пределах нашей задачи основывается на возможности хранения данных в одном файле, удобство использования и возможность расширения.

Работы с базой ведётся через отдельную программу под названием `SqliteDog`, она разработана для действенного управления базами данных SQLite. Она подсвечивает синтаксис и разрешает дополнять имена базы, помимо прочего имеет сильный редактор схем БД, позволяющий загрузить её практически сразу. Дополнительно ко всему `SqliteDog` имеет возможность перестраивать схему безопасно внутри транзакции и интеллектуально подбирать ширину столбцов данных.

3 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ КОМПЬЮТЕРНОЙ РЕКОНСТРУКЦИИ

После рассмотрения собранных данных о главных силах японского и российского флотов, а также изучения хронологии боя, перейдем непосредственно к самому процессу построения компьютерной реконструкции Цусимского сражения. В данной главе рассматриваются вопросы синтеза собранных чертежей и фотографий боевых кораблей в программе трехмерного моделирования Blender, конечным результатом которого станет построение компьютерной реконструкции Цусимского сражения.

3.1 Создание 3D-моделей кораблей

Трехмерное моделирование дает возможность реалистично создавать абсолютно фантастические проекты или напротив, точно воспроизводить реальный мир. В данном разделе будет приведено поэтапное описание создания интерактивной трехмерной модели боевого корабля, на примере эскадренного броненосца «Сисой Великий» – реально существовавшего объекта. Каждый этап моделирования снабжен необходимыми комментариями и примерами экранных форм.

Создание 3D моделей боевых кораблей является важным этапом для будущей виртуальной реконструкции, так как необходимо реализовать модель корабля с минимальным количеством полигонов и максимальной детализированностью.

3.1.1 Описание инструментов и этапов моделирования

Перед тем как непосредственно перейти к самому моделированию необходимо собрать и проанализировать как можно больше информации о моделируемом корабле. На основании собранных ранее чертежей размеры кораблей записывалась для точного соблюдения пропорций.

Моделирование корабля можно разделить на два этапа:

- 1) непосредственно создание основы корабля;
- 2) текстурирование модели с последующим созданием развёртки тексту-

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		48

ры.

Рассмотрим оба этапа более подробно.

Трёхмерное изображение на плоскости отличается от двумерного тем, что включает построение геометрической проекции трёхмерной модели сцены на плоскость (например, экран компьютера) с помощью специализированных программ [2]. При этом модель может, как соответствовать объектам из реального мира (автомобили, здания, ураган, астероид), так и быть полностью абстрактной (проекция четырёхмерного фрактала).

В следующем разделе будет подробно описано, как создавалась трёхмерная модель эскадренного броненосца «Сисой Великий».

3.1.2 Создание полигональной модели

При моделировании происходит постоянное дробление полигональной сетки с целью получения большей детализации объекта. Легче всего поделить полигон четырехугольной формы, потому что при моделировании она является предпочтительной и наиболее удобной. При импорте модели в большинство графических процессоров полигональная сетка автоматически разбивает четырехугольники на треугольники. Далее на основе высокополигональной модели создается низкополигональная. На этом этапе важно соблюдать правильность топологии и избегать многоугольников.

Создание основы модели корабля условно можно разделить на два этапа:

- 1) создание основы конструкций корабля;
- 2) создание мелких элементов конструкций корабля.

На рисунке 26 представлен эскадренный броненосец «Сисой Великий», на примере которого мы рассмотрим оба этапа моделирования.

На первом этапе происходило построение рабочего чертежа модели корабля, понимаемого программой Blender 2.70, в которой, впоследствии, осуществлялось построение трёхмерных моделей боевых кораблей. Чертежи были приведены в единый масштаб, после чего происходила интеграция данных чертежей и текстур непосредственно в программу Blender.

Далее в рабочем окне программы Blender, на рабочей сцене в проекциях

Top, Left, Front были установлены изображения соответствующих проекций чертежа корабля.



Рисунок 26 – Модель эскадренного броненосца «Сисой Великий»

Затем на сцену был добавлен примитив Plane и растянут таким образом, чтобы его границы совпадали с границами борта корабля. После этого, в местах изгиба линии борта были добавлены новые рёбра [43]. Результат представлен на рисунке 27.

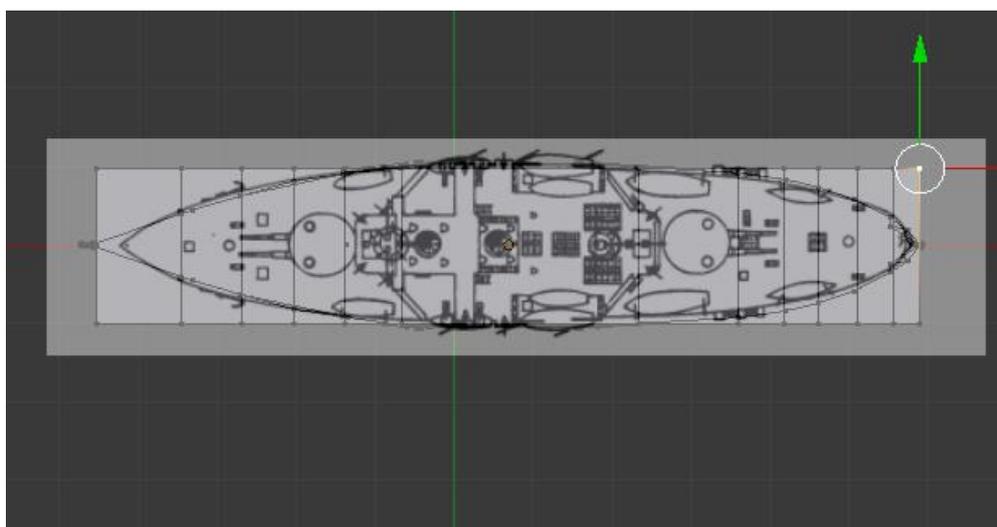


Рисунок 27 – Результат создания примитива Plane

Затем узловые точки каждого ребра были совмещены с линией борта корабля. Таким образом, мы получили элемент, повторяющий очертания кораб-

ля. Далее, при помощи инструмента Extrude, полученный ранее элемент был выдавлен до киля корабля, в местах изгиба бортов были вставлены дополнительные рёбра. Узловые точки рёбер были совмещены с линией борта в поперечных сечениях корабля. В итоге был получен корпус корабля, представленный на рисунке 28.

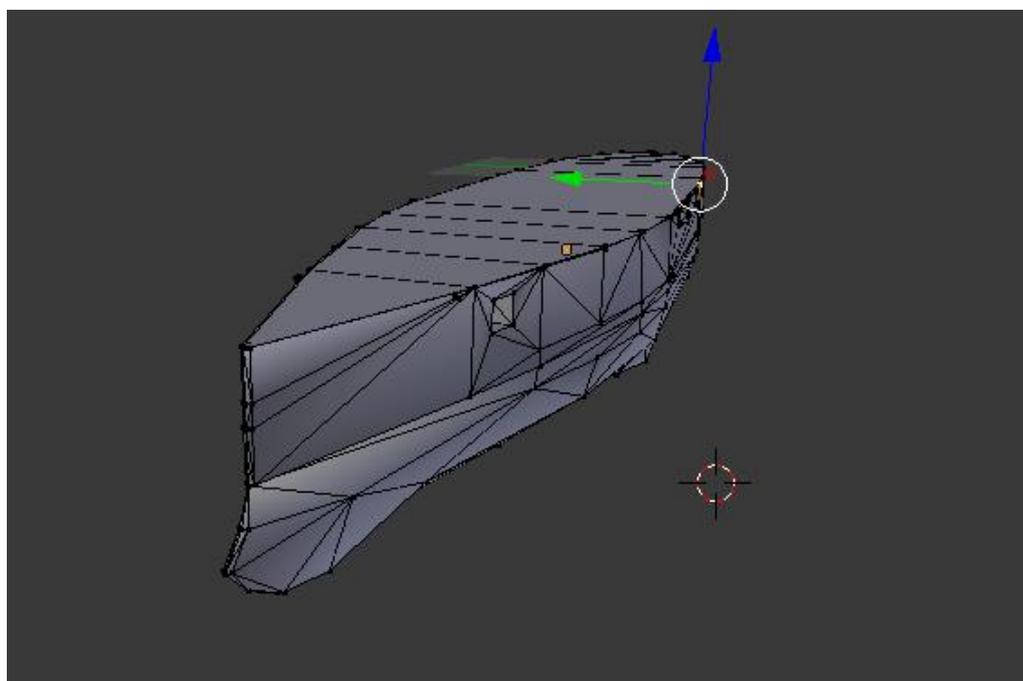


Рисунок 28 – Корпус броненосца «Сисой Великий»

Аналогичным образом к корпусу были добавлены каземат и боевая рубка. При помощи примитива Circle и инструмента Extrude были созданы и добавлены к общей конструкции орудийные башни, артиллерийские орудия, мачты, дымовые трубы, вентиляционные шахты. Необходимо учесть то, что требуется подбирать оптимальные методы моделирования, чтобы впоследствии модель имела как можно меньше полигонов в своей структуре. На этом первый этап разработки модели был завершён.

На втором этапе разработки по сохранившимся фотографиям и чертежам были созданы различные мелкие элементы конструкции, такие как якоря, иллюминаторы, устройства для крепления шлюпок и другие элементы. Результат выполнения второго этапа представлен на рисунке 29.

Так как среда Unity3D имеет ограничение по количеству полигонов трех-

мерной модели (поддержка до 65000 полигонов геометрии), необходимой процедурой являлось построение низкополигональной трехмерной модели корабля с помощью инструментов ретопологии. После этого количество полигонов было снижено, однако при этом, качество и детализация модели остались почти без изменения.



Рисунок 29 – Полигональная модель броненосца «Сисой Великий»

3.1.3 Реализация синтетической текстуры

Текстуры – это сделанные вручную или программно-сгенерированные изображения, которые накладываются на поверхность моделей для детализации. Чтобы соотнести вершины геометрии и пиксели плоского изображения необходимо создать текстурную развертку или UV-развертку.

UV-развёртка (англ. UV map) – это соответствие между координатами на поверхности трёхмерного объекта (X, Y, Z) и координатами на текстуре (U, V). [28] Создание UV-развертки подразумевает под собой разрезание низкополигональной модели на отдельные куски и размещение их в пространстве координат UV.

Blender содержит отдельный модуль для работы с материалами и текстурами. Все манипуляции с текстурами осуществляются на панели Textures окна

Properties.

Текстуры нельзя соединять с моделью без наличия материала. Для этого, вначале добавляем материал, и лишь потом создаем текстуры.

Материалом называется набор настроек, описывающий свойства поверхности. Материал можно назначить на любой объект, но нельзя использовать как задний фон. Для изменения параметров материалов необходимо открыть соответствующую вкладку на панели инструментов, расположенной справа (рисунок 30).

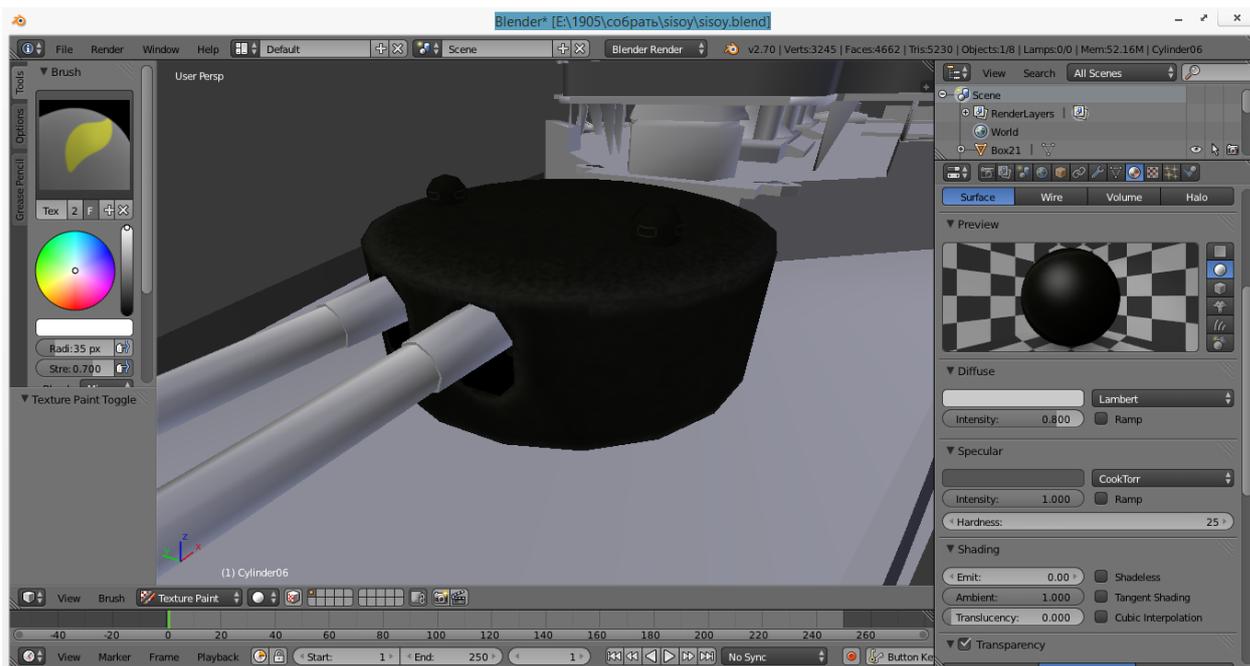


Рисунок 30 – Настройка отображения материалов оружейной башни

Вверху отображается список созданных материалов. В процессе создания материала ему присваивается имя, а также настраиваются параметры отражения света, прозрачности, отбрасываемых теней, существует возможность цветокоррекции и тонирования.

Текстуры были подготовлены заранее на основе изображений и чертежей.

В случае с методом моделирования, применяемым для компьютерной реконструкции, все детали смоделированы в виде объемной геометрии. Таким образом, нужны текстуры, имитирующие фактуру разнообразных материалов и текстуры, несущие информацию об освещенности.

Качество поверхности текстуры определяется ее разрешением, то есть количеством пикселей на минимальную единицу текстуры. Созданные нами, UV двумерные координаты, в последствии были перенесены на текстуру. Для этого мы задействовали встроенный редактор UV Editor. Результат проекции текстуры на модель корабля были отредактированы с помощью опций группы Image Mapping.

На вкладке Colors находится набор инструментов, с помощью которых нами был откорректирован цвет, яркость и контрастность разработанной текстуры.

Корректировка яркости, контрастности и насыщенности осуществлялась в области параметров Adjust. Помимо этого, настройка цветового баланса изображения была произведена с помощью стандартной регулировки RGB.

В итоге мы получаем реалистичную модель корабля с минимальным количеством полигонов и реалистичной текстурой [50].

Конечный результат представлен на рисунке 31.



Рисунок 31 – Итоговая модель броненосца «Сисой Великий»

3.2 Создание редактора сценариев

В основе построения компьютерной реконструкции лежит принцип при-

ближенного воспроизведения изучаемого хода боя; имитируются составляющие его элементарные явления с сохранением их логической структуры и последовательности протекания во времени [35, 36].

Важное направление разработки реконструкции хода боя – это воссоздание динамики происходящего сражения, вплоть до мельчайших деталей.

На начальных этапах разработки системы рассматривались два варианта воспроизведения хода боя: создание скриптовых сцен или создание полноценного редактора сценариев. Взвесив все за и против, выбор был сделан в пользу последнего, так как данный метод обеспечивает гибкость при создании сцены боя, большую скорость работы и возможность нелинейного воспроизведения сценария.

Реализации данного метода состояла из нескольких стадий: разработка базы данных для хранения информации о кораблях и сценарии реконструкции, разработка модуля создания сценария и модуля – интерпретатора сценариев.

Сценарий представляет собой упорядоченную по времени последовательность действий, таких как: изменение направления движения, скорости, целей для орудий корабля, включение и выключение звуковых и визуальных эффектов.

Для удобства навигации, сценарий был разделен на эпизоды. Каждый эпизод содержит в себе начальные координаты, предназначенные для первоначальной расстановки кораблей на карте. Для этого нами был разработан соответствующий блок, выполняемый в начале каждого эпизода. По завершению работы данного блока происходит обнуление локального времени t . Далее управление передается модулю – интерпретатору сценариев. Алгоритм работы программы представлен на рисунке 32.

Рассмотрим более подробно блок «Воспроизведение эпизода». Данный блок, представляющий собой интерпретатор сценариев, выполняет разбор входящей последовательности и преобразует ее, в понятные графическому редактору, команды по взаимодействию объектов на сцене. Таким образом, осуществляется воспроизведение заданного сценария боя.

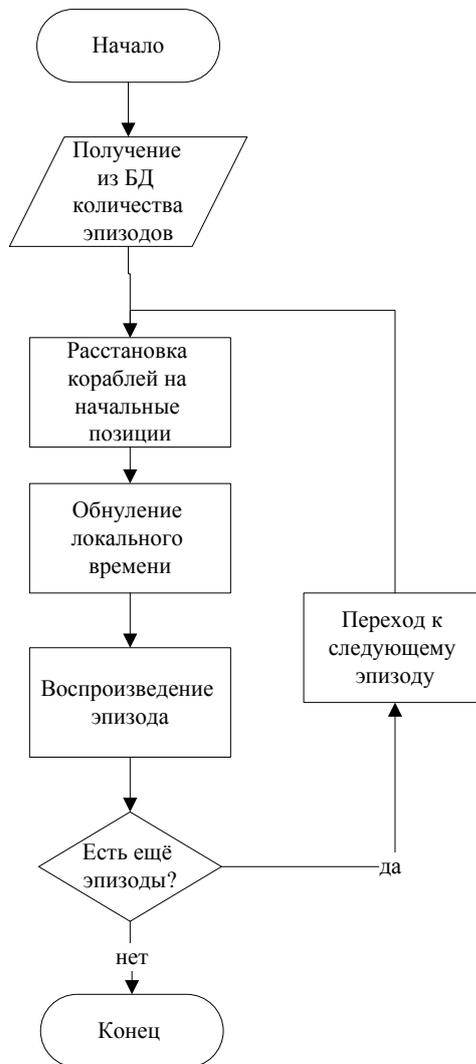


Рисунок 32 – Блок-схема алгоритма работы программы

Для хранения информации о кораблях нами был написан класс «ships», содержащий в себе информацию о курсе, скорости, цели и состоянии орудий, активных визуальных эффектов для каждого корабля.

Для каждого действия на сцене, такого как смена курса, скорости корабля, поворота башни, выстрела, в базе данных содержится запись, характеризующая это действие. Для реализации поставленных задач нами был разработан алгоритм, представленный на рисунке 33.

Исходя из вышесказанного, суть работы данного алгоритма сводится к следующему:

1) в базе данных производится проверка на наличие непрочитанных записей, если такие записи имеются, то переходим к шагу 2, иначе шаг 5;

- 2) производится считывание n-го количества записей и переход к шагу 3;
- 3) производится проверка: настало ли время выполнить следующую задачу, если да, то переходим к шагу 4, иначе к шагу 3.
- 4) выбранная задача запускается на выполнение, и параллельно ей запускается проверка наличия следующей незапущенной задачи, если такая имеется, то переходим к шагу 3, иначе к шагу 1;
- 5) производится ожидание завершения работы всех запущенных задач и завершение работы алгоритма.

Листинг интерпретатора сценариев представлен в приложении Б.

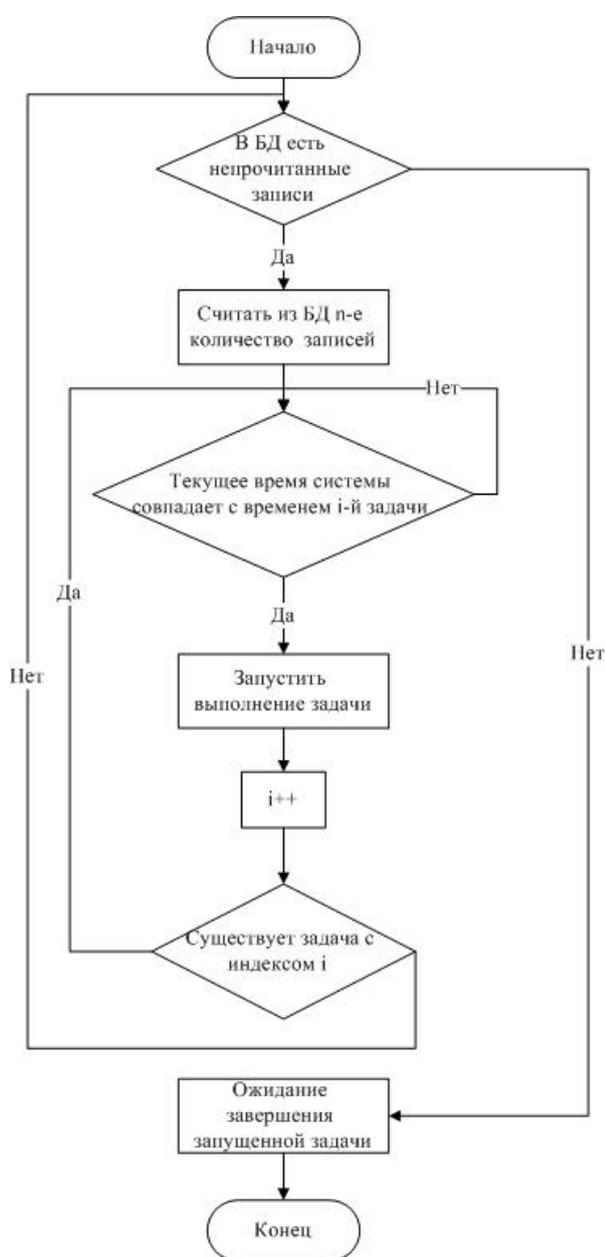


Рисунок 33 – Блок-схема работы интерпретатора сценариев

Итогом работы алгоритма является интерактивный фильм с возможностью прокрутки времени просмотра, его замедление и ускорение. Прокрутка времени осуществляется путем выбора воспроизводимого эпизода. Также донесения образа исторического сражения осуществляется в системе с помощью реалистичных визуальных и звуковых эффектов, а также возможности изменения угла и точки обзора. Это достигается благодаря предоставлению пользователю возможности переключения между камерами, закрепленными за объектами и свободной камерой.

3.3 Проектирование интерфейса программного модуля

Основной задачей на момент начала создания пользовательского интерфейса являлась правильная компоновка клавишей управления, ведь все объекты на сцене должны быть максимально четко видны пользователю.

Поэтому было принято решение о расположения меню по периметру окна просмотра.

Следующим этапом стал поиск практических решения с учетом возможностей программной среды. Местом для просмотра моделей станет заранее подготовленная сцена. Меню выбора будет располагаться над сценой с возможностью перемещения над поверхностью.

Этап практической реализации начался с создания сцены.

При запуске Unity3d мы автоматически получаем пустую сцену с необходимым набором инструментов, для её наполнения и работы со скриптами. Работа в данной программе проста и интуитивно понятна. Начальная сцена представлена на рисунке 34.

В верхней области программы мы будем наблюдать непосредственно тот вид, который будет представлен в будущем пользователю. Две нижних области отвечают за импортированные модели и скрипты, область справа в дальнейшем позволит нам настроить проект [42].

Для придания окружающей атмосфере более естественного вида и реалистичности используется Skybox – шестисторонний куб, который прорисовывается позади всей остальной графики.

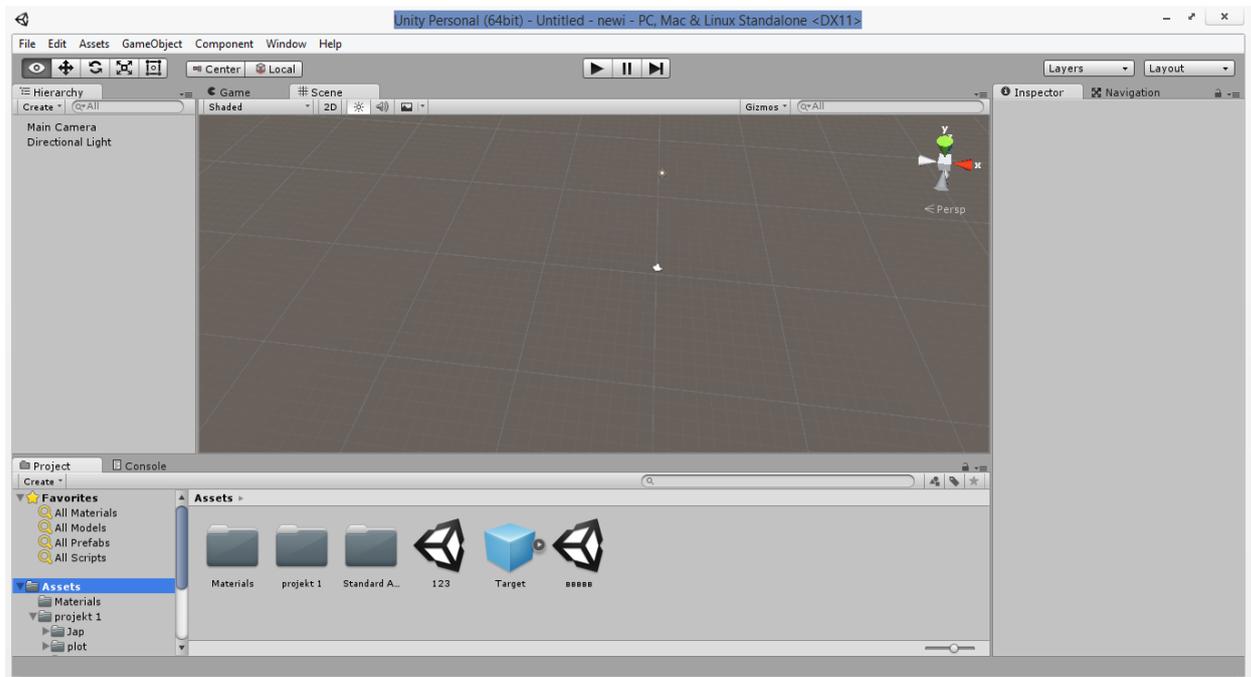


Рисунок 34 – Стартовое окно Unity3D

Шаги, необходимые для создания Skybox:

- 1) сделать шесть текстур, которые соответствуют каждой из шести сторон Skybox и положить их в папку Assets;
- 2) для каждой текстуры требуется изменить режим обертки (Wrap Mode) с Repeat на Clamp. Если этого не сделать, то цвета по краям не будут совпадать;
- 3) создать новый материал, выбрав Assets\Create\Material в строке меню выбрать шейдер из выпадающего списка в верхней части Inspector, затем выбрать RenderFX->Skybox;
- 4) назначить шесть текстур для каждого слота текстуры в материале.
Можно сделать это путем перетаскивания каждой текстуры из окна ProjectView на соответствующие слоты.

Для привязки скайбокса к сцене, необходимо сделать следующее:

- выбрать Edit\Render Settings из главного меню;
- перетащить материал скайбокса в слот Skybox Material в Inspector.

В случае нашего использования Skybox были задействованные готовые варианты, так как текстура неба есть в стандартных наборах. Дальнейшим действием на сцену был добавлен эффект воды, также встроенный в опции Unity3d.

Чтобы придать нужный цвет поверхности обратимся снова к панели модификаций, и выберем кисть с подходящей текстурой.

Чтобы модели, появляющиеся на сцене, можно было без труда разглядеть необходимо правильно расставленное освещение. Для этого прибегнем к эффекту освещения - Lights. В Unity3d, источник света можно использовать для освещения сцены или объектов, создавая тем самым особое визуальное настроение. Свет можно использовать для моделирования солнца, света от горящей спички, ручного фонарика, выстрелов из оружия, или взрывов, и многие другие варианты в которых можно применить источники света.

В Unity3d есть четыре разновидности источников света:

- Directional lights данный объект располагается бесконечно далеко и освещает все объекты сцены. Эффект очень похож на эффект солнечного света;
- Point lights светит одинаково во всех направлениях создавая эффект лампочки;
- Spot lights светит из точки в определенном направлении, освещая объекты попадающие внутрь конуса свечения, данный эффект схож с эффектом автомобильных фар;
- Area lights светит во всех направлениях на одной стороне прямоугольной области плоскости.

Все вышеперечисленные источники света могут породить Shadows. Свойства теней могут быть настроены для каждого источника света в отдельности.

Для освещения создаваемой в данном проекте сцены наиболее подходящим будет источник света Directional lights. Чтобы выбрать его снова обратимся к верхней панели меню GameObject затем lights\ Directional lights.

Результатом выполнения вышеуказанных действий стала трехмерная сцена, для вывода моделей изображенная на рисунке 35.

Следующим этапом в создания пользовательского интерфейса являются камеры. Принцип использования камер в Unity3d, подобен принципу использования камер в фильмах.

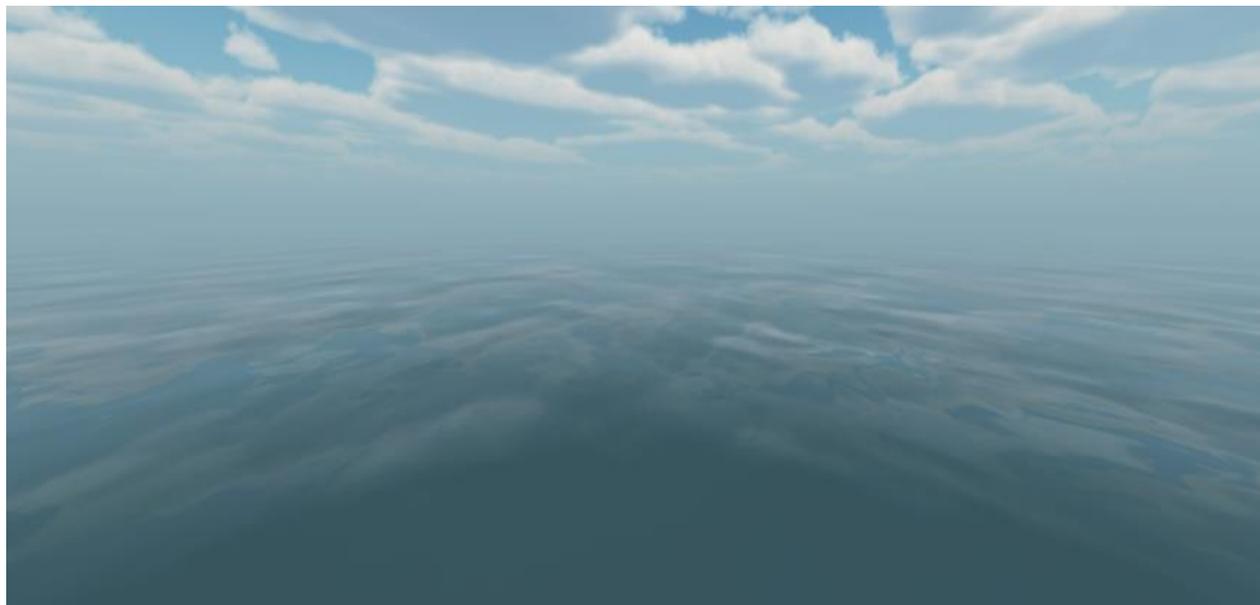


Рисунок 35 – Трехмерная сцена для вывода моделей

Существует возможность анимировать камеры либо управлять ими согласно всем законам физики. С помощью камер можно получить практически любое изображение, можно использовать стандартные и уникальные камеры, которые больше подходят стилю проекта.

Для реализации данного проекта была задействована одна камера с возможностью орбитального вращения вокруг модели.

В Unity3D, начиная с версии 4.6, встроен удобный набор инструментов для создания гибкого пользовательского интерфейса. Главным элементом пользовательского интерфейса является Canvas – контейнер для хранения набора UI. Он может вмещать любое количество элементов. Все вновь создаваемые элементы интерфейса добавляются в Canvas, вне элемента Canvas создать элемент интерфейса невозможно.

Следующим этапом в создании пользовательского интерфейса стало добавление кнопок управления скоростью воспроизведения и смены камер. Для обработки нажатий на кнопки интерфейса был написан скрипт на языке C#. Написание скриптов происходит во встроенном редакторе Unity3D.

В процессе разработки скрипт не однократно тестируется, визуализатор в случае не правильного написания скрипта выдаёт нам ошибку, что позволяет быстро найти место ошибки и исправить её, тем самым мы повышаем надёж-

ность разрабатываемой программы.

Итоговый вид пользовательского интерфейса представлен на рисунке 36.



Рисунок 36 – Итоговый вид пользовательского интерфейса

3.4 Проектирование базы данных

Следующим шагом проектирования программного модуля является проектирование базы данных.

Проектирование баз данных проходило в три этапа [5]:

- инфологическое проектирование – выделение сущностей и назначение им атрибутов;
- логическое проектирование – построение логической структуры базы данных, приведение отношений к нормальным формам;
- физическое проектирование – описываются таблицы в том виде, в котором они реализованы средствами СУБД.

На этапе инфологического проектирования был определён набор основных сущностей необходимых для полноценной работы БД. Опишем эти сущности [20]:

- 1) сущность «Корабль» содержит информацию о боевых кораблях русского и японского флотов;

2) сущность «Класс корабля» содержит информацию о принадлежности корабля к тому или иному классу боевых надводных кораблей;

3) сущность «Сторона» содержит информацию о принадлежности корабля к той или иной противоборствующей стороне;

4) сущность «Сценарий» содержит информацию о ходе создаваемого боя;

5) сущность «Эпизод» содержит информацию о фрагментах сюжета хроники боя;

6) сущность «Тип действия» содержит информацию об структурных единицах деятельности корабля, относительно завершенных отдельных актов.

Далее выполним выбор идентифицирующих атрибутов и назначение сущностям описательных атрибутов; так как путём анализа предметной области были выделены следующие наборы атрибутов для каждой сущности, представленные ниже в таблицах 4–9.

Все атрибуты имеют свои спецификации, а сущности идентифицируются ключами. Идентифицирующие атрибуты подчеркиваются сплошной линией.

Таблица 4 – Атрибуты сущности «Корабль»

Название атрибута	Описание атрибута	Диапазон значений	Пример
<u>Код_корабля</u>	Идентификатор, однозначно определяющий номер корабля	До 2 цифр	2
Название	Наименование боевого корабля	До 30 символов	Сисой Великий

Таблица 5 – Атрибуты сущности «Класс корабля»

Название атрибута	Описание атрибута	Диапазон значений	Пример
<u>Код_класса</u>	Идентификатор, однозначно определяющий принадлежность класса корабля	До 1 цифры	1
Класс	Информация о принадлежности корабля к классу боевых надводных кораблей	До 50 символов	Эскадренный броненосец

Таблица 6 – Атрибуты сущности «Сторона»

Название атрибута	Описание атрибута	Диапазон значений	Пример
<u>Код стороны</u>	Идентификатор, однозначно определяющий противоборствующую сторону	До 1 цифры	1
Сторона конфликта	Информация о противоборствующей стороне	До 30 символов	Россия

Таблица 7 – Атрибуты сущности «Сценарий»

Название атрибута	Описание атрибута	Диапазон значений	Пример
<u>Код события</u>	Идентификатор, однозначно определяющий то или иное события	До 3 цифр	3
Время	Информация о времени возникновения события	До 7 цифр	129
P1	Параметр события N	До 5 цифр	37,2
P2	Параметр события N	До 5 цифр	0
P3	Параметр события N	До 5 цифр	1
P4	Параметр события N	До 5 цифр	23
P5	Параметр события N	До 5 цифр	20,3

Таблица 8 – Атрибуты сущности «Эпизод»

Название атрибута	Описание атрибута	Диапазон значений	Пример
<u>Код эпизода</u>	Идентификатор, однозначно определяющий фрагмент сюжета хроники боя	До 2 цифр	2
Название	Информация о фрагменте сюжета хроники боя	До 30 символов	Гибель эскадренного броненосца «Бородино»

Таблица 9 – Атрибуты сущности «Тип действия»

Название атрибута	Описание атрибута	Диапазон значений	Пример
<u>Код типа</u>	Идентификатор, однозначно определяющий тип произошедшего действия	До 2 цифр	2
Описание	Информация об структурных единицах деятельности корабля, относительно завершенных отдельных актах	До 40 символов	Поворот башни корабля

В процессе концептуально – инфологического проектирования БД были установлены связи между сущностями [10]. Все сущности, связанные между собой типом связи «один-ко-многим».

Результатом данного этапа инфологического проектирования базы данных является концептуальная инфологическая модель в виде диаграммы «сущность-связь», представленная на рисунке 37.

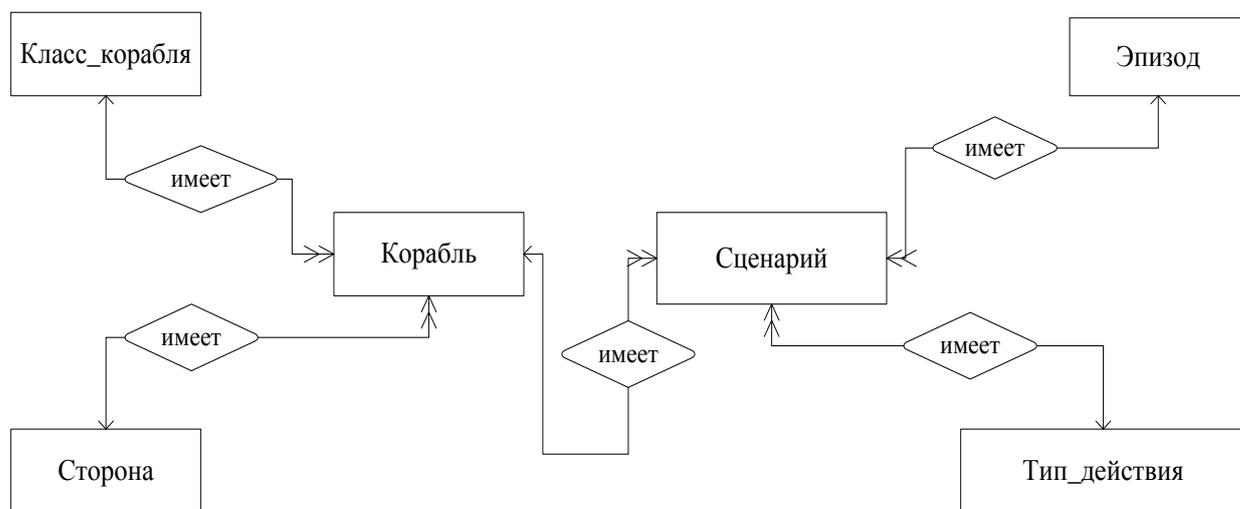


Рисунок 37 – Концептуальная инфологическая модель

На этапе логического проектирования было осуществлено построение реляционной логической модели. Реляционно – логическая модель – совокупность нормализованных отношений, в которой реализованы связи между объектами ПО и выполнены все преобразования, необходимые в конкретной среде организации СУБД [21]. В результате данного этапа, после отображения концептуальной инфологической модели на логическую реляционную модель и

анализа полученных отношений с точки зрения соответствия их трём нормальным формам, мы выделили дополнительные ассоциативные сущности «Детали перелома» и «Детали компоновки» ассоциативная и получили реляционно-логическую модель БД. В приложении В.1 приведена логическая модель базы данных. Эта модель построена с помощью AllFusion Erwin Data Modeller и содержит отношения, приведенные к трем нормальным формам (дальнейшая нормализация отношений считается не целесообразной) [22].

В заключении проектирования БД, на этапе физического проектирования, представляются проекты таблиц, которые будут непосредственно реализованы в СУБД SQLite. В приложении В.2 приведена физическая модель базы данных.

В результате физического проектирования была создана диаграмма базы данных, на которой отражены все сущности, атрибуты, связи между сущностями, а также первичные и внешние ключи (рисунок 38).

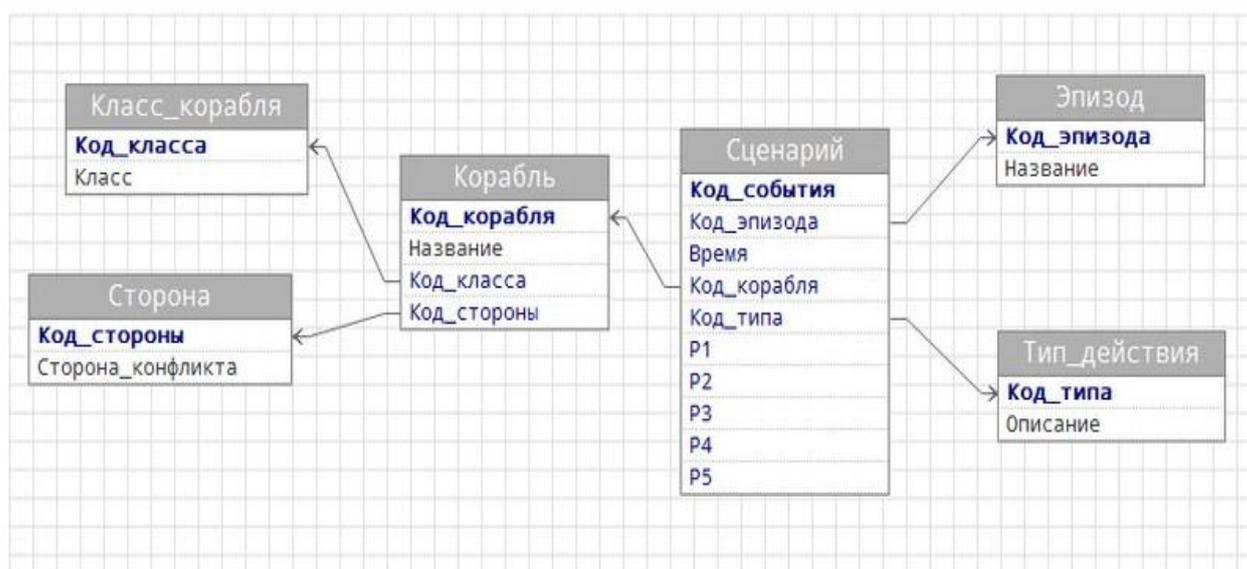


Рисунок 38 – Диаграмма базы данных

3.5 Руководство пользователя

Запуск программы осуществляется посредством двойного щелчка левой кнопкой мыши по исполняемому файлу «1905.exe», расположенному в каталоге \1905.

По завершению загрузки запустится окно конфигурации графики, в котором будет предложено выбрать разрешение экрана и качество графики, а также

режим отображения: в окне или на весь экран.

Окно конфигурации графики представлено на рисунке 39.

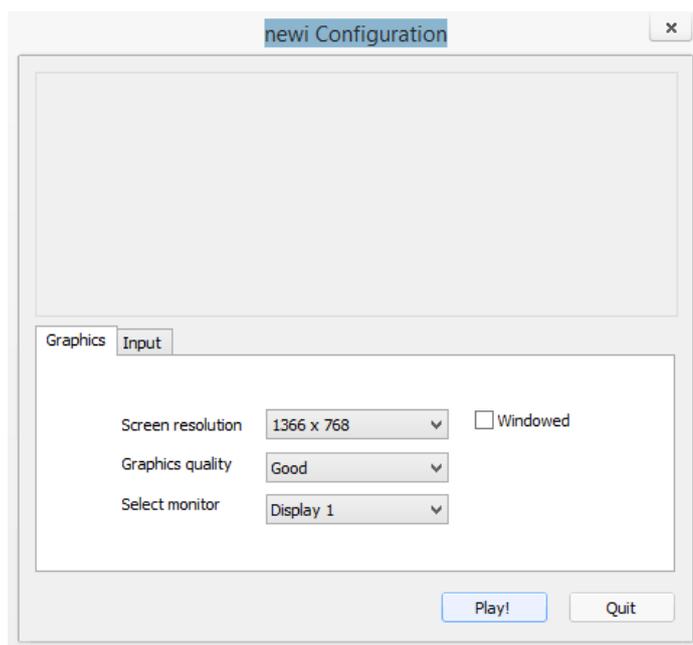


Рисунок 39 – Окно конфигурации графики

Так мы можем запускать приложение на разных рабочих станциях в зависимости от мощности и требований пользователя. Для первого запуска рекомендуется запускать в режиме Good. Выставив нужные параметры, нажимаем кнопку Play после чего откроется приложение и начинается воспроизведение боя.

В приложении предусмотрена возможность смены камеры, для более детального рассмотрения интересующих боевых кораблей. Также имеется свободная камера, которая позволяет перемещаться над театром боевых действий без привязки к определенному кораблю. Для перемещения свободной камеры в пространстве используются клавиши «W», «S», «A», «D».

Смена ракурса осуществляется с помощью компьютерной мыши, для этого необходимо просто удерживать среднюю клавишу мыши и перемещать курсор, вращение камеры осуществляется при зажатой правой клавише мыши.

Кнопки «+» и «-», расположенные на экранном интерфейсе, позволяют замедлять или ускорять ход боя, кнопки «<<» и «>>» осуществляют переключение эпизодов боя (рисунок 40).

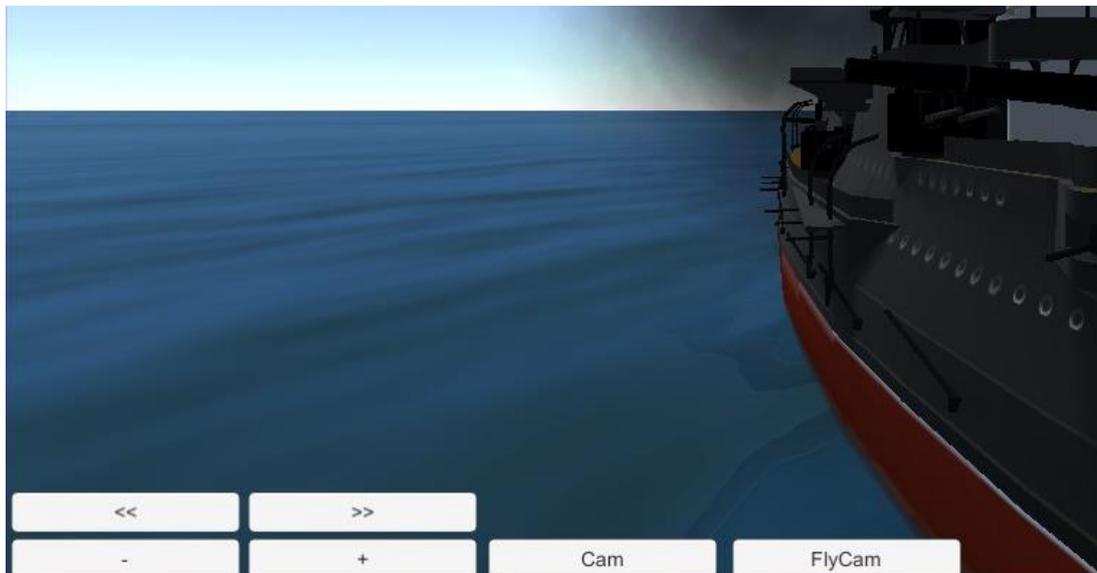


Рисунок 40 – Итоговый вид пользовательского интерфейса

Пример работы программы показан на рисунке 41.



Рисунок 41 – Пример работы программы

Завершение работы программы осуществляется посредством нажатия кнопки «Выход», расположенной в левом нижнем углу рабочего окна.

Изм.	Лист	№ докум.	Подп.	Дата

ВКР.155518.09.04.04.ПЗ

Лист

68

ЗАКЛЮЧЕНИЕ

Объектом исследования диссертационной работы выступает крупнейшее в эпохе додредноутного броненосного флота Цусимское сражение.

Целью диссертационной работы является повышение интереса к объектам историко-культурного наследия, путем создания виртуальной реконструкции Цусимского сражения.

Научная новизна результатов:

1) для боевых кораблей противоборствующих сторон Цусимского сражения созданы трехмерные модели, учитывающие их реальные пропорции и максимальную детализацию реально существовавших прототипов;

2) предложен метод построения редактора сценариев, осуществляющий создание сцены боя;

3) разработан программный продукт, представляющий собой компьютерную реконструкцию Цусимского сражения.

Практическое значение и реализация результатов работ.

Разработанный программный продукт, представляющий собой интерактивный фильм в формате виртуальной реальности, позволяет достичь эффекта проникновения человека (пользователя системы) в историческую среду, что в свою очередь способно вызывать эмоциональный отклик у зрителя. Благодаря зрелищности, наглядности и высокому уровню детализации, виртуальная реконструкция поможет привлечь внимание пользователей всех возрастов и любого уровня образования к вопросам истории, а также может использоваться в образовательной деятельности, в задачах развития историко-культурного туризма.

Итоговая система получилась универсальной, что позволяет легко адаптировать ее для применения в других предметных областях.

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		69

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 Баранов, Ю. М. Реконструкция утраченных промышленных объектов и раритетных технологий с использованием компьютерного моделирования / Ю. М. Баранов, Е. А. Курлаев // Российский научно-технический музей проблемы и перспективы. – Нижний Тагил, 2000. – С. 46-53.

2 Большаков, В. П. Инженерная и компьютерная графика. Теоретический курс и тестовые задания: учеб. пособие / В. П. Большаков. – СПб. : БХВ-Петербург, 2016. – 384 с.

3 Бородкин, Л. И. Моделирование исторических процессов: от реконструкции реальности к анализу альтернатив / Л. И. Бородкин. – СПб. : Алетей, 2016. – 304 с.

4 Быков, П. Д. Сражение у о. Цусима / отв. ред. Р. Н. Мордвинов. – М., 1951. – С. 348–367.

5 Грабер, М. SQL. Справочное руководство / М. Грабер. – 2-е изд. – М. : Лори, 2011. – 354 с.

6 Груздев, Д. В. Визуализация и анализ результатов археолого- географических исследований: задачи, режимы и программная реализация / Д. В. Груздев, И. В. Журбин // Информационный бюллетень Ассоциации «История и компьютер». – М., 2006. – № 33. – С. 43-52.

7 Виртуальная реконструкция объектов историко-культурного наследия / М. В. Румянцев [и др.]. – М. : Синергия, 2011. – № 5.

8 Всероссийский научно-методический семинар «Виртуальная реконструкция в форматах научного исследования и учебного процесса» / под ред. М. В. Румянцева, Л. И. Бородкина, М. А. Лаптевой // Научный журнал Сибирского федерального университета Сибирский федеральный университет, Красноярск, 2011. – №7. – Т4. – С. 139-144.

9 Галенин, Б. Г. Описание военных действий на море в 37–38 гг. Мэйдзи: От Владивостока до Цусимы / Б. Г. Галенин. – М., 2004. – С. 183-194.

10 Диго, С. М. Базы данных. Проектирование и создание / С. М. Диго. –

					ВКР.155518.09.04.04.ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		70

М. : Изд-во ЕАОИ, 2012. – 171 с.

11 Житин, Р. М. Виртуальная реконструкция комплекса хозяйственных и жилых построек Ново-Покровского имения / Р. М. Житин // Социально-экономические явления и процессы. – 2014. – № 9. – С. 113-119.

12 Жеребятьев, Д. И. Методы трёхмерного компьютерного моделирования в задачах исторической реконструкции монастырских комплексов Москвы / Д. И. Жеребятьев. – М. : МАКС Пресс, 2014. – 224 с.

13 Жеребятьев, Д. И. Применение технологий интерактивного 3D моделирования для реконструкции утраченных памятников истории и архитектуры на примере крепости Тамбов / Д. И. Жеребятьев // мат. XIV междунар. конф студентов, аспирантов и молодых ученых «Ломоносов-2007» – М. : Изд-во МГУ, 2007. – С. 214.

14 Заключение следственной комиссии по выяснению обстоятельств Цусимского боя : офиц. текст // Морской сборник. –1917. – 57 с.

15 Заменит ли виртуальный Парфенон оригинал? [Электронный ресурс]. – Режим доступа : http://news.bbc.co.uk/hi/russian/entertainment/newsid_3561000/3561907.stm – 14.02.2017.

16 История русско-японской войны 1904-1905 гг. / под ред. И. И. Ростунова. – М., 1977. – С. 324-348.

17 Колисниченко, Д. Н. PHP 5/6 и SQLite. Разработка Web-приложений / Д. Н. Колисниченко. – 2-е изд. – СПб. : БХВ-Петербург, 2015. – 560 с.

18 Кончаков, Р. Б. Технологии трехмерного моделирования в ракурсе исторической информатики / Р. Б. Кончаков, Д. И. Жеребятьев. – М. : Изд-во МГУ, 2010. – С. 145-175.

19 Копейкин, М. В. Базы данных. Основы SQL реляционных баз данных: учеб. пособие. / М. В. Копейкин, В. В. Спиридонов, Е. О. Шумова. – СПб. : СЗТУ, 2014. – 177 с.

20 Мишанкина, Н. А. Базы данных в лингвистических исследованиях / Н. А. Мишанкина // Вопр. Лексикографии, 2013. – №3. – С.25-33.

21 Мирошниченко Г. Ф. Реляционные базы данных : практические прие-

					ВКР.155518.09.04.04.ПЗ	Лист
Изм.	Лист	№ докум.	Подп.	Дата		71

мы оптимальных решений / Г.Ф. Мирошниченко. – СПб. : Изд-во БХВ-Петербург, 2013. – 400 с.

22 Назарова, О. Разработка реляционных баз данных с использованием CASE-средства All Fusion Data Modeler / О. Назарова, О. Масленникова. – М. : ЛитРес, 2015. – 130 с.

23 Озеров, К. В. Правда о броненосце «Сисой Великий» (Цусимское сражение) / К. В. Озеров, Г. В. Озеров. – М. : ООО «Горизонт», 2016. – 376 с.

24 Пиков, Н. О. Виртуальная реконструкция Спасского мужского монастыря г. Енисейска с применением технологии Quest3D / Н. О. Пиков // Молодежь и наука : сб. мат. VI всерос. научн.-техн. конф студентов, аспирантов и молодых ученых – Красноярск : Сиб. федер. ун-т, 2011.

25 Пискунов, А. Г. Руководство по SQLite для пользователей Windows : [Электронный ресурс] / А. Г. Пискунов. – Режим доступа : <http://agp1.hx0.ru/.SQLite.Allow.pdf>. – 29.10.2016.

26 Полутов, А. В. К вопросу о возможности прорыва эскадры адмирала Рожественского через пролив Цугару / А. В. Полутов // Россия и АТР. – 2009. – №4. – С. 162-170.

27 Прахов, А. А. Blender. 3D-моделирование и анимация / А. А. Прахов. – М. : Изд-во Феникс, 2015. – 256 с.

28 Прахов, А. А. Blender: 3D-моделирование и анимация. Руководство для начинающих / А. А. Прахов. – СПб. : БХВ-Петербург, 2009. – 272с.

29 Прахов, А. А. Самоучитель Blender 2.6. / А. А. Прахов. – СПб. : БХВ-Петербург, 2013. – 384 с.

30 Самойлова, Е. О. Проблема времени в виртуальном нарративе компьютерных игр / Е. О. Самойлова // Философские проблемы информационных технологий и киберпространства. – 2012. – №2. – С. 107-113.

31 Смирнов, М. И. Цусима (Сражение в Корейском проливе 14–15 / V 1905.) / М. И. Смирнов. – СПб., 1913. – 305 с.

32 Смит, Б. Исследовательская деятельность Европейского Союза в области культурного наследия / Б. Смит // VI межд. конф. EVA. – 2003. – 17 с.

					<i>ВКР.155518.09.04.04.ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		72

33 Смолин, А. А. Виртуальная реконструкция храмового комплекса г. Енисейска на основе технологии трехмерного моделирования / А. А. Смолин, М. В. Румянцев // Круг идей: Методы и технологии исторических реконструкций. – 2010. – С. 175-196.

34 Соловов, А. В. Проектирование компьютерных систем учебного назначения / А. В. Соловов. – Самара : СГАУ, 1995. – 137 с.

35 Сороговец, А. В. Особенности компьютерной реконструкции Цусимского сражения / А. В. Сороговец, В. В. Еремина // Вестник Амурского государственного университета. 2017. – Вып. 75 – С. 50-52.

36 Сороговец, А. В. Компьютерная реконструкция Цусимского сражения / А. В. Сороговец // Молодёжь XXI века: шаг в будущее : мат. XVII рег. научн.-пр. конф. – Благовещенск : БГПУ, 2016. – Т. 3. – С.242–244.

37 Степанов, А. Н. Информатика: Учебник для вузов / А. Н. Степанов. – 5-е изд. – СПб. : Питер, 2015. – 720 с.

38 Сулига, С. В. Корабли русско-японской войны : Справочное издание / С. В. Сулига. – М. : Дскольдъ, 1993. – 104 с.

39 Троелсен, Э. С# и платформа .NET. Библиотека программиста : пер. с англ. / Э. Троелсен. – СПб. : Питер, 2014. – 796 с.

40 Усачев, А. В. Концепция информационной системы «Актуализация историко-культурного наследия» / А. В. Усачев, М. В. Румянцев, Р. А. Барышев // Прикладная информатика. – 2011. – № 2. – С. 55-68.

41 Фищев, А. В. Реконструкция исторического прошлого в виртуальной среде компьютера / А. В. Фищев // Вопросы информатизации образования. – 2009. – № 13.

42 Фленов, М. Е. Искусство программирования игр на С# / М. Е. Фленов. – СПб. : БХВ-Петербург, 2014. – 256 с.

43 Фореев, В. Н. Компьютерная графика/ В.Н. Фореев. – СПб. : БХВ-Петербург, 2012. – 432 с.

44 Хомоненко, А. Д. Базы данных / А. Д. Хомоненко. – СПб. : Корона принт, 2012. – 414 с.

45 Чаплыгин, В. А. Цусимское сражение: ход боя / В. А. Чаплыгин // Электронный научный журнал Курского государственного университета. – 2011. – №3. – С. 182-188.

46 Часовский, П. В. Семиотика «Easter eggs», или игровое начало в компьютерных играх / П. В. Часовский // Вестник Челябинского государственного университета. – 2012. – №36. – С. 63-66.

47 Энциклопедия Кругосвет [Электронный ресурс]. – Режим доступа : http://www.krugosvet.ru/enc/istoriya/RUSSKO-YAPONSKAYA_VONA.html. – 26.03.2017.

48 Яблоков, К. В. Исторические компьютерные игры как способ моделирования исторической информации / К. В. Яблоков. – М., 2007. – С. 170-204.

49 Яблоков, К. В. Компьютерные исторические игры 1990 - 2000-х гг. : Проблемы интерпретации исторической информации : дис. . канд. ист. наук / К. В. Яблоков ; РГГУ – Москва, 2005. – 252 с.

50 Blender Basics 3-rd edition [Электронный ресурс]. – Режим доступа : http://www.b3d.mezon.ru/index.php/Blender_Basics_3-rd_edition. – 06.04.2017.

51 Conway's All the World's Fighting Ships, 1860-1905. – London : Conway Maritime Press, 1979.

52 Gerasimov, V. Unity 3.x Scripting / V. Gerasimov, D. Kraczla. – Packt Publishing, 2012. – 292 с.

53 Goldstone, W. Unity 3.x Game Development Essentials, Game development with C# and Javascript / W. Goldstone. – Packt Publishing, 2011. – 488 с.

54 Le plan de Rome. Restituer la Rome antique. The Scale Model of Rome [Электронный ресурс]. – Режим доступа : http://www.unicaen.fr/services/cireve/rome/pdr_maquette.php?fichier=histoire. – 12.02.2017.

ПРИЛОЖЕНИЕ А

ЦУСИМСКОЕ СРАЖЕНИЕ 14–15 (27–28) мая 1905 г.

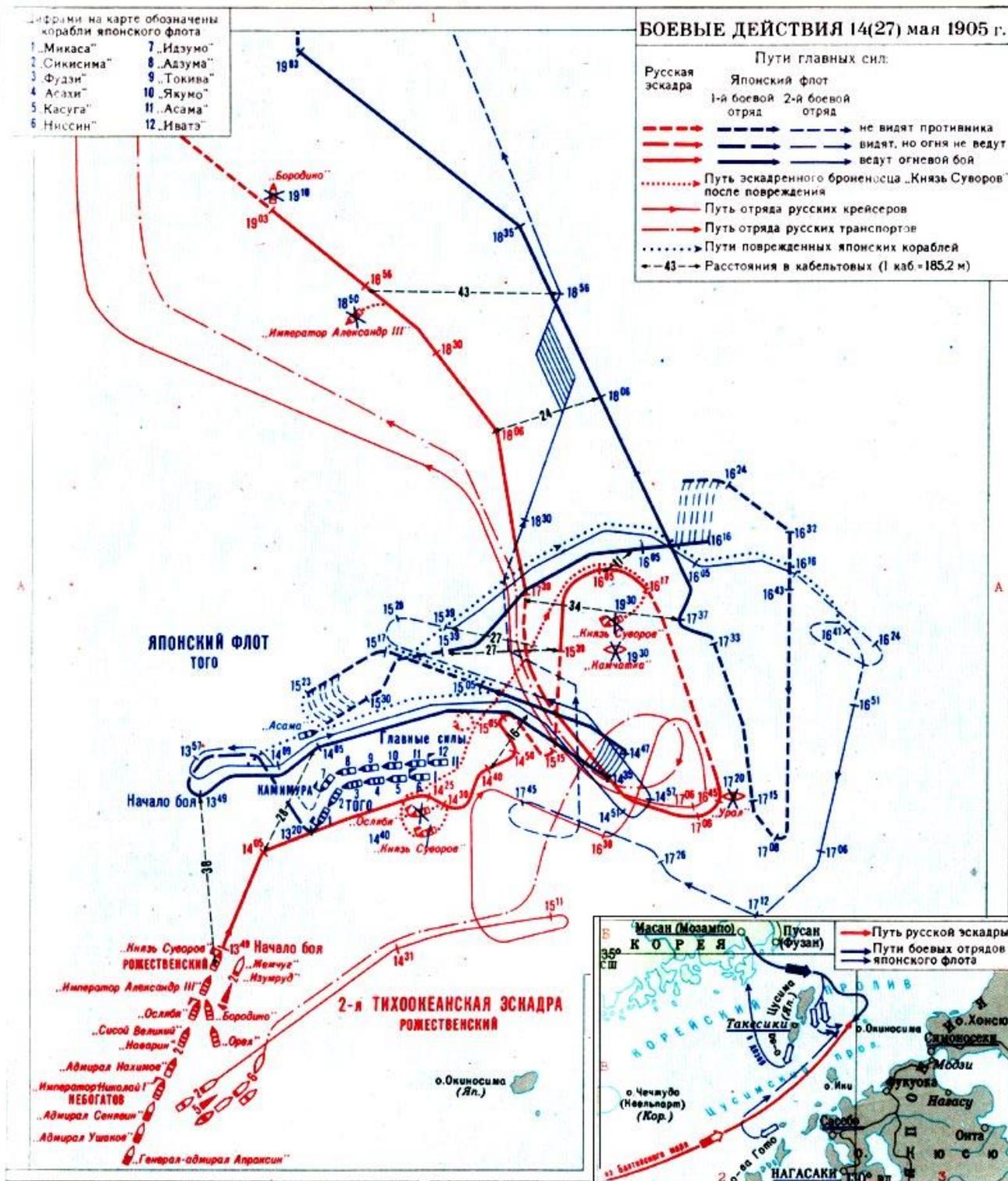


Рисунок А.1 – Карта боевых действий Цусимского сражения

ПРИЛОЖЕНИЕ Б

Листинг интерпретатора сценариев

```
void Update()
{ //формируем задания на движение и т.д
  if (fl)
  {
    if (i < kl && targetPozM[i, 0] <= Time.time-l_time)
    { //устанавливаем курс
      if (targetPozM[i, 2] == 1)
      {
        Destroy(GameObject.Find("temp" + targetPozM[i, 1]));
        GameObject instance;
        instance = Instantiate(target, new Vector3(targetPozM[i, 3], target-
PozM[i, 4], targetPozM[i, 5]), new Quaternion(0, 0, 0, 0)) as GameObject;
        instance.name = "temp" + targetPozM[i, 1];
        int t = Mathf.RoundToInt(targetPozM[i, 1]);
        ship[t].targetPoz = instance;
        ship[t].speed = targetPozM[i, 6];
        ship[t].speed_p = targetPozM[i, 7];
      }
    }
    else //устанавливаем цель для орудий
    if (targetPozM[i, 2] == 2)
    {
      int t = Mathf.RoundToInt(targetPozM[i, 1]);
      ship[t].target_gk = Mathf.RoundToInt(targetPozM[i, 3]); }
    else //производим выстрел
    if (targetPozM[i, 2] == 3)
    { int z;
      int t = Mathf.RoundToInt(targetPozM[i, 1]);
      for (int h =3; h < 8; h++)
```

Продолжение ПРИЛОЖЕНИЯ Б

Листинг скрипта отображения кораблей

```
{ z = Mathf.RoundToInt(targetPozM[i, h]);
  if (z != 0 && ship[t].bg_GK[h-3]==1)
  { ship[t].gk_shot[z-1].SetActive(false);
    ship[t].gk_shot[z-1].SetActive(true); } }
}
else //включаем эффекты
  if (targetPozM[i, 2] == 4)
  { int t = Mathf.RoundToInt(targetPozM[i, 1]);
    int h = Mathf.RoundToInt(targetPozM[i, 3]);
    int e = Mathf.RoundToInt(targetPozM[i, 4]);
    if (e == 0)
    { ship[t].spec_effect[h].SetActive(false); }
    else
      if (e == 1)
        { ship[t].spec_effect[h].SetActive(true); } }
  else //изменяем состояние орудий
    if (targetPozM[i, 2] == 5)
    { int t = Mathf.RoundToInt(targetPozM[i, 1]);
      int h;
      for(int n=3;n<8;n++)
        { h = Mathf.RoundToInt(targetPozM[i, n]);
          ship[t].bg_GK[n-3] = h; } }
    //переходим к следующей записи
    i++; }
//выполняем задания
int x = ship.GetLength(0);
//двигаем корабль
for (int j = 0; j < x; j++)
```

Изм.	Лист	№ докум.	Подп.	Дата

Продолжение ПРИЛОЖЕНИЯ Б

Листинг скрипта отображения кораблей

```
{ if (ship[j].targetPoz)
    { ship[j].trl = Vector3.Lerp(ship[j].trl, ship[j].targetPoz.transform.position,
Time.deltaTime * ship[j].speed_p);
    ship[j].ship.transform.LookAt(ship[j].trl); // смотрим на текущую цель-
вейпоинт
    // двигаемся к текущей цели-вейпоинту
    ship[j].ship.transform.position = Vector3.MoveTowards
(ship[j].ship.transform.position,ship[j].targetPoz.transform.position, ship[j].speed);
    } }
//наводим орудия
for (int k = 0; k < x; k++)
    { int y = ship[k].gk.GetLength(0);
      for (int j = 0; j < y; j++)
        { int ind = ship[k].target_gk;
          if (ship[k].bg_GK[j] == 1)
            { ship[k].n_gk[j] = Vector3.MoveTowards(ship[k].n_gk[j],
ship[ind].ship.transform.position, 1f);
            ship[k].gk[j].transform.LookAt(ship[k].n_gk[j]); // смотрим на текущую цель
            } } } }
```

ПРИЛОЖЕНИЕ В

Модель БД

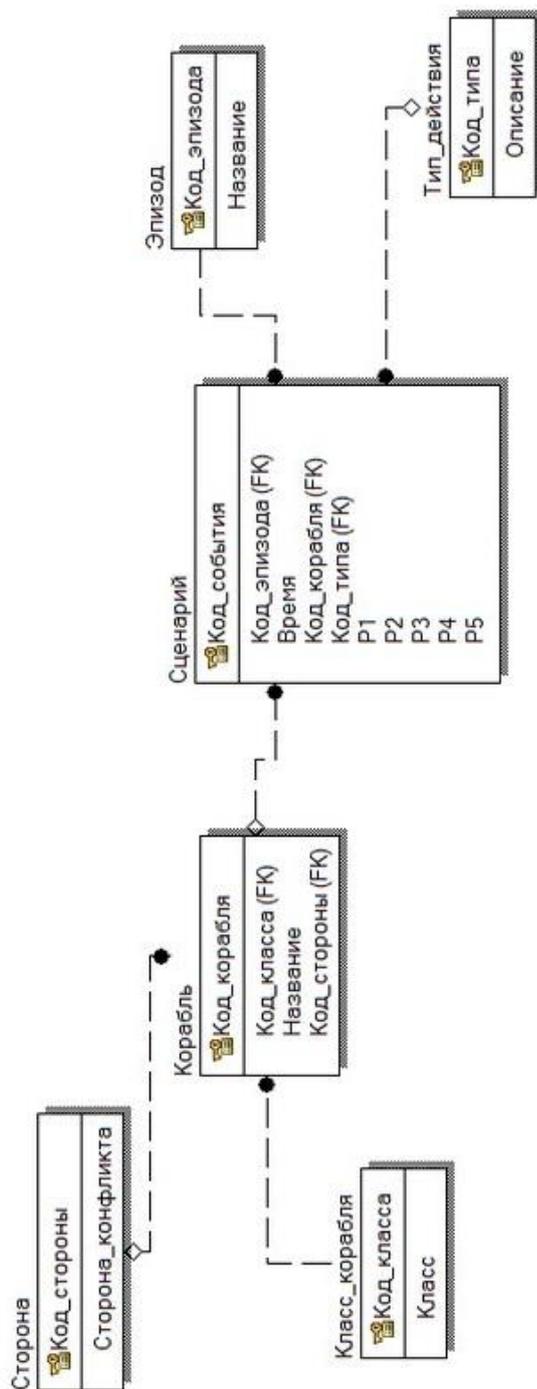


Рисунок В.1 – Логическая модель базы данных

Продолжение ПРИЛОЖЕНИЯ В

Модель БД

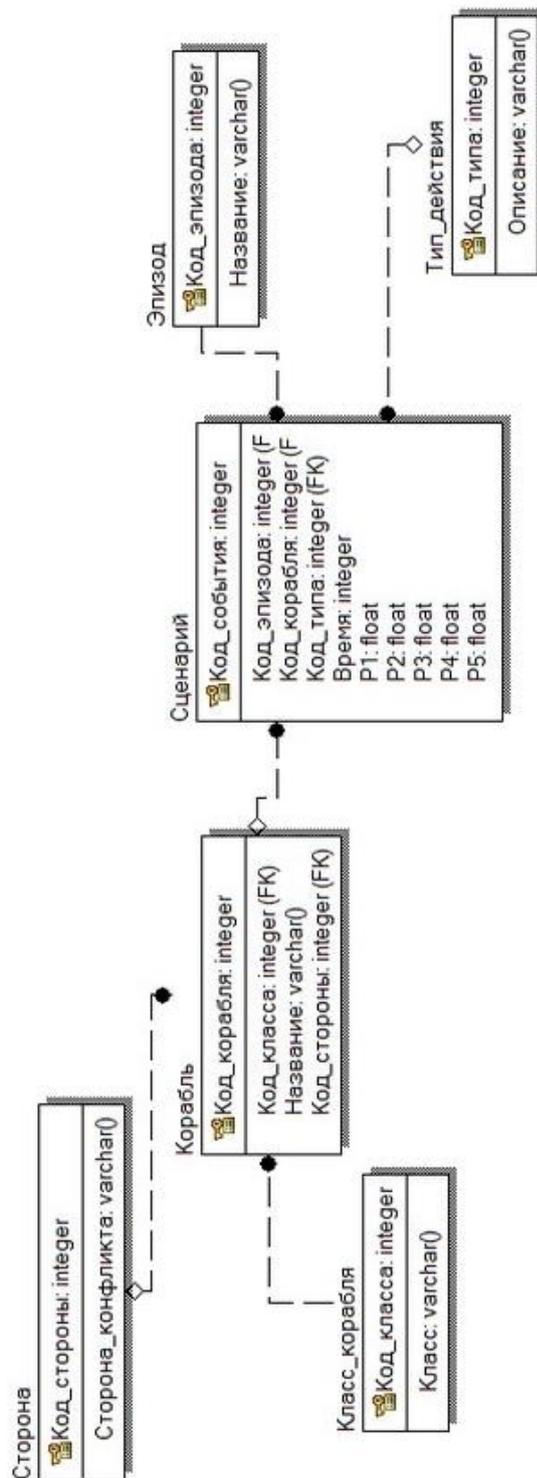


Рисунок В.2 – Физическая модель базы данных