

Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Курсовой проект
по предмету «Проектирование архитектур ПО»

ИНФОРМАЦИОННАЯ СИСТЕМА «ФОНДОВАЯ БИРЖА»

Работу выполнил студент гр. 53501/3 _____ Мартынов С. А.

Работу принял преподаватель _____ Зозуля А. В.

Санкт-Петербург
2016

Содержание

Введение	4
1 Функциональные требования	5
1.1 Роли	5
1.2 Функциональные требования по ролям	5
1.2.1 Администратор биржи	5
1.2.2 Трейдер	6
1.2.3 Анонимный пользователь	6
1.2.4 Регулятор	6
1.3 Прочие требования	7
2 Варианты использования	8
2.1 Администратор биржи	8
2.1.1 Авторизация в программе	8
2.1.2 Регистрация нового трейдера	9
2.1.3 Редактирование трейдера	9
2.1.4 Блокирование трейдера	9
2.1.5 Регистрация новой ценной бумаги	9
2.1.6 Редактирование ценной бумаги	10
2.1.7 Блокирование ценной бумаги	10
2.1.8 Задание рамок торговой сессии	10
2.1.9 Задание рамок и условий аукциона открытия	11
2.1.10 Задание рамок и условий основных торгов	11
2.1.11 Задание рамок и условий аукциона закрытия	12
2.1.12 Блокирование работы биржи	12
2.2 Трейдер	13
2.2.1 Авторизация в программе	13
2.2.2 Получение статуса работы биржи	13
2.2.3 Получение списка торгуемых инструментов	14
2.2.4 Получение рыночной информации	14
2.2.5 Подача заявки на покупку/продажу инструмента	14
2.2.6 Отзыв заявки на покупку/продажу инструмента	14
2.2.7 Получение торгового баланса	15
2.3 Анонимный пользователь	15
2.3.1 Получение статуса работы биржи	15
2.3.2 Получение списка торгуемых инструментов	15

2.3.3	Получение рыночной информации	16
2.4	Регулятор	17
2.4.1	Авторизация в программе	17
2.4.2	Получение статуса работы биржи	17
2.4.3	Получение рыночной информации	18
2.4.4	Блокирование работы биржи	18
2.4.5	Блокирование трейдера	18
2.4.6	Блокирование ценной бумаги	19
3	Статическая модель предметной области	20
4	Динамическая модель предметной области	24
5	Архитектура приложения	25
5.1	Бизнес логика	25
5.2	Инфраструктурный слой	26
5.3	Сервисный слой	27
5.4	Слой представления	27
5.5	Слой веб-сервисов	29
6	Тестирование	30
	Заключение	31
	Список литературы	32

Введение

Информационная система представляет собой сильно упрощенную модель фондовой биржи.

Задачами фондовой биржи в частности является:

- Предоставление централизованного места, где может происходить как продажа ценных бумаг их первым владельцам, так и вторичная их перепродажа;
- Выявление равновесной биржевой цены;
- Обеспечение гласности, открытости биржевых торгов;
- Установление правил торговли;
- Обеспечение арбитража;
- Обеспечение гарантий исполнения сделок;
- Аккумулирование временно свободных денежных средств и способствование передаче права собственности;
- Разработка этических стандартов, кодекса поведения участников биржевой торговли.

Централизованным местом будет разрабатываемая информационная система, предоставляющая трейдерам возможность выставить заявки на покупку и продажу инструментов (ценных бумаг). Процесс торговли неизбежно будет формировать выявление равновесной цены. Для обеспечения прозрачности всех процедур, доступ к торговой информации может иметь любой желающий, в то время как проводить сделки (выставлять заявки) могут только авторизованные трейдеры. Трейдер может осуществлять любые сделки в соответствии со своей торговой стратегией, однако пространство арбитража ограничено одной торговой площадкой, таким образом возможна спекуляция только по времени (при этом биржа естественно не отслеживает портфель инвестора, т.е. он может выставить на продажу сколько угодно бумаг). Администрация биржи осуществляет управление биржей по средствам установки торговых правил (допуск инструментов, гранулярность) и гарантирует исполнение торговых поручений (с точностью до секунды), в соответствии с правилами. Помимо администрации самой биржи, надзор за торгами имеет ряд контролирующих органов.

За пределами нашего рассмотрения окажутся вопросы аккумулирования денежных средств и передачи прав собственности т.к. это требует построения значительно более сложной модели. Этические стандарты и кодексы поведения так же не будет рассмотрена.

1 Функциональные требования

Разработать информационную систему «Фондовая биржа», позволяющую организовать процесс продажи и покупки ценных бумаг участниками рынка (трейдерами) с соблюдением требований регулирующих органов.

1.1 Роли

Информационная система должна обладать следующими ролями:

Администратор биржи – осуществляет управление работой биржи, добавляя/удаляя трейдеров и ценные бумаги.

Трейдер – осуществляет торговлю на бирже, выставя заявки на продажу и покупку ценных бумаг, отслеживает торговый баланс в рамках торговой сессии.

Анонимный пользователь – подписывается на рыночную информацию с биржи (чаще всего это строение сервисы, занимающие накопление и анализом рыночных данных для прогнозирования цен).

Регулятор – осуществляет надзор за работой биржи и может приостановить торги в соответствии со своими внутренними правилами (не всегда эти правила публично доступны а блокировка происходит в случае возникновения подозрений о махинациях или опасных изменениях цены каких-то инструментов).

1.2 Функциональные требования по ролям

Каждой роле в информационной системе присущи собственные функциональные требования.

1.2.1 Администратор биржи

- авторизация;
- регистрация нового трейдера (администратор выдаёт логин и пароль);
- редактирование (в т.ч. изменение статуса блокировки, выставленного администратором) существующего трейдера;
- регистрация торгового инструмента (ценной бумаги), определение гранулярности торговли;

- редактирование (в т.ч. изменение статуса блокировки, выставленного администратором) торгового инструмента (ценной бумаги);
- задание/изменение рамок торговой сессии: аукциона открытия, основных торгов, аукциона закрытия а так же доли биржи от совершения операций;
- получение/редактирование статуса блокировки биржи (выставляемого администратором);

1.2.2 Трейдер

- авторизация;
- получение статуса работы биржи;
- получение рыночной информации в рамках торговой сессии и дальнейшие обновления;
- получение списка торгуемых инструментов;
- подача/отзыв заявки на покупку/продажу инструмента (включая вид инструмента, цену, объём с учётом гранулярности, срок исполнения);
- получение торгового баланса в рамках торговой сессии.

1.2.3 Анонимный пользователь

- получение статуса работы биржи;
- получение списка торгуемых инструментов;
- получение рыночной информации в рамках торговой сессии и дальнейшие обновления.

1.2.4 Регулятор

- авторизация;
- получение рыночной информации в рамках торговой сессии и дальнейшие обновления;
- получение/редактирование статуса блокировки биржи (выставляемого регулятором);
- получение/редактирование статуса блокировки ценной бумаги (выставляемого регулятором);
- получение/редактирование статуса блокировки брокера (выставляемого регулятором);

1.3 Прочие требования

К прочим существенным требованиям относятся

1. проведение всех операций, подразумевающих создание/редактирование данных требует авторизации;
2. проведение сделок осуществляется по правилу наименьшей цены;
3. точность исполнения заявок – 1 секунда;
4. отсутствие айсберг-заявок;
5. блокировки могут быть срочными и бессрочными;
6. блокировки администратора и регулятора независимы и работают по правилам дизъюнкции.

2 Варианты использования

Варианты использования представлены на диаграммах прецедентов (Use Case Diagrams).

2.1 Администратор биржи

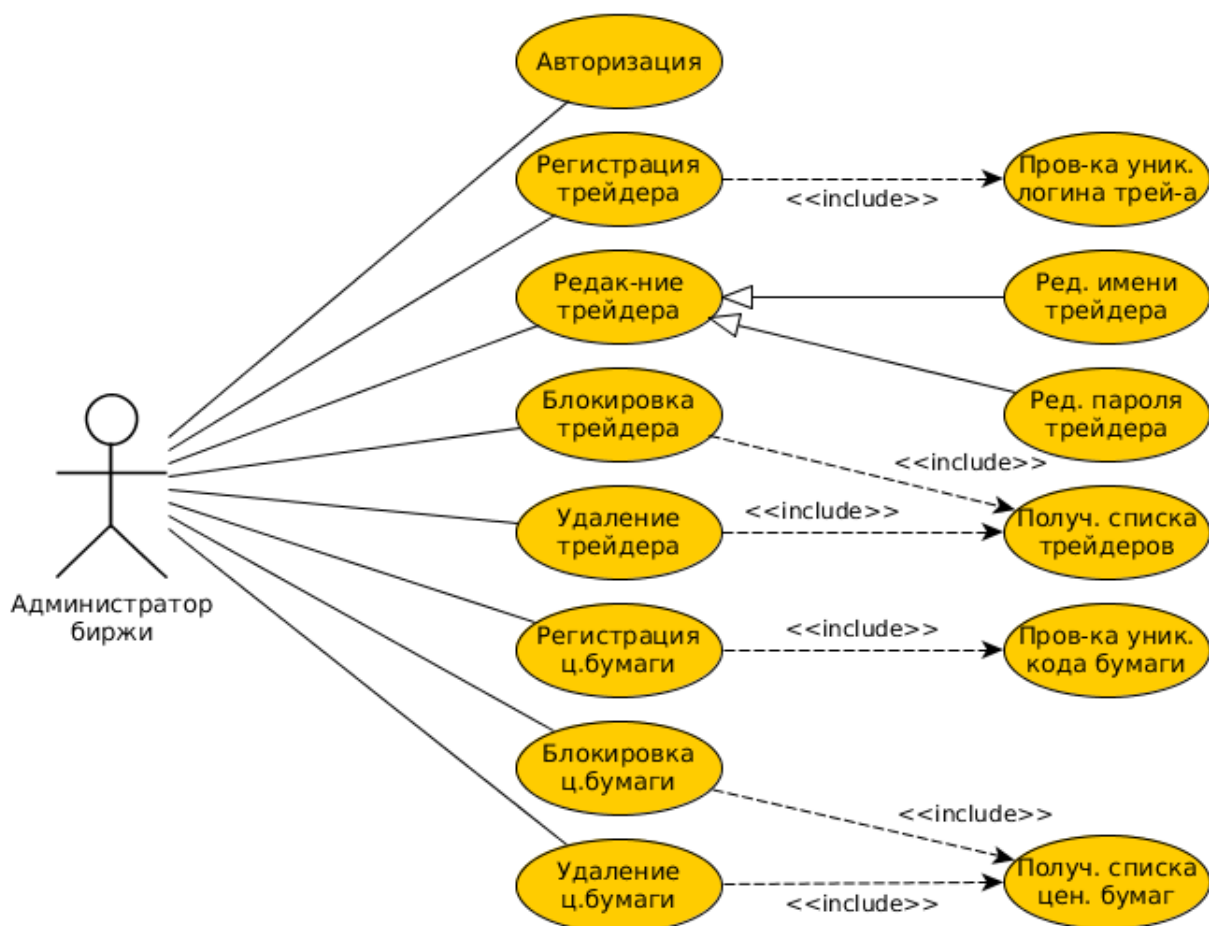


Рис. 1: Диаграммах прецедентов администратора биржи

2.1.1 Авторизация в программе

1. Для того чтобы выполнить вход в систему под пользователем Администратор биржи, необходимо ввести логин и пароль в специальной форме при запуске программы.

Алт.1 Если Пользователь не может выполнить авторизацию как Администратор биржи, он может продолжить работу как Анонимный пользователь.

2.1.2 Регистрация нового трейдера

1. Администратор биржи вводит данные нового трейдера в специальном окне и сохраняет запись
 - (Обязательно) Уникальный логин
 - (Обязательно) Пароль (генерируется администратором)
 - (Обязательно) Имя физического или юридического лица
 - (Опционально) примечания

Алт.1 Если администратор биржи не ввёл какое либо из полей или логин не является уникальным, то он получает сообщение об ошибке а новый трейдер не считается созданным.

2.1.3 Редактирование трейдера

1. Администратор биржи получает список трейдеров
2. Администратор изменяет пароль и/или имя и/или примечание трейдера
3. Администратор сохраняет изменения

2.1.4 Блокирование трейдера

1. Администратор биржи получает список трейдеров
2. Администратор выставляет срочную блокировку трейдера (указывая дату и время когда блокировка будет снята на любой момент в будущем).
3. Администратор сохраняет изменения

Алт.2 Администратор выставляет бессрочную блокировку трейдера (выставляя дату и время снятия блокировки в 0).

Алт.2 Если ранее имелась блокировка, выставленная Администратором, то он может её снять (выставляя дату и время снятия блокировки в любое значение, предшествующее текущему моменту, но не 0).

2.1.5 Регистрация новой ценной бумаги

1. Администратор биржи вводит данные новой ценной бумаги в специальном окне и сохраняет запись

- (Обязательно) Уникальный код (как правило, 3-5 букв латинского алфавита)
- (Обязательно) гранулярность торгов
- (Опционально) примечания

Алт.1 Если уникальный код не является уникальным, то Администратор получает сообщение об ошибке а новая ценная бумага не считается созданной.

2.1.6 Редактирование ценной бумаги

1. Администратор биржи получает список ценных бумаг
2. Администратор изменяет гранулярность торгов и/или примечание
3. Администратор сохраняет изменения

2.1.7 Блокирование ценной бумаги

1. Администратор биржи получает список ценных бумаг
2. Администратор выставляет срочную блокировку ценной бумаги (указывая дату и время когда блокировка будет снята на любой момент в будущем).
3. Администратор сохраняет изменения

Алт.2 Администратор выставляет бессрочную блокировку ценной бумаги (выставляя дату и время снятия блокировки в 0).

Алт.2 Если ранее имелась блокировка, выставленная Администратором, то он может её снять (выставляя дату и время снятия блокировки в любое значение, предшествующее текущему моменту, но не 0).

2.1.8 Задание рамок торговой сессии

1. Администратор биржи в специальном окне задаёт время начала и окончания торговой сессии
2. Администратор сохраняет изменения

Алт.2 Если время окончания торговой сессии совпадает или предшествует времени начала, то Администратор получает сообщение об ошибке а данные не сохраняются.

2.1.9 Задание рамок и условий аукциона открытия

1. Администратор биржи в специальном окне отмечает, должны ли применяться правила аукциона открытия
2. Администратор задаёт время окончания аукциона открытия (время начала совпадает с временем начала торговой сессии, а время окончания не может превышать время окончания торговой сессии; время аукциона открытия не должно пересекаться со временем основных торгов и временем аукциона закрытия)
3. Администратор отмечает галочкой, должна ли биржа получать процент с каждой сделки во время аукциона открытия и выставляет этот процент (0 или более)
4. Администратор отмечает галочкой, должна ли биржа получать фиксированную сумму с каждой сделки во время аукциона открытия и выставляет эту фиксированную сумму в копейках (0 или более)
5. Администратор сохраняет изменения

Алт.5 Если введенные данные нарушают ограничения, то Администратор получает сообщение об ошибке а данные не сохраняются.

2.1.10 Задание рамок и условий основных торгов

1. Администратор биржи в специальном окне отмечает, должны ли применяться правила основных торгов
2. Администратор не задаёт время основных торгов (время начала должно превышать время начала торговой сессии, а время окончания не может превышать время окончания торговой сессии; время основных торгов не должно пересекаться со временем аукциона открытия и временем аукциона закрытия)
3. Администратор отмечает галочкой, должна ли биржа получать процент с каждой сделки во время основных торгов и выставляет этот процент (0 или более)
4. Администратор отмечает галочкой, должна ли биржа получать фиксированную сумму с каждой сделки во время основных торгов и выставляет эту фиксированную сумму в копейках (0 или более)
5. Администратор сохраняет изменения

Алт.5 Если введенные данные нарушают ограничения, то Администратор получает сообщение об ошибке а данные не сохраняются.

2.1.11 Задание рамок и условий аукциона закрытия

1. Администратор биржи в специальном окне отмечает, должны ли применяться правила аукциона закрытия
2. Администратор задаёт время открытия аукциона закрытия (время начала должно превышать время начала торговой сессии, а время окончания совпадает со временем окончания торговой сессии; время аукциона закрытия не должно пересекаться со временем аукциона открытия и временем основных торгов)
3. Администратор отмечает галочкой, должна ли биржа получать процент с каждой сделки во время аукциона закрытия и выставляет этот процент (0 или более)
4. Администратор отмечает галочкой, должна ли биржа получать фиксированную сумму с каждой сделки во время аукциона закрытия и выставляет эту фиксированную сумму в копейках (0 или более)
5. Администратор сохраняет изменения

Алт.5 Если введенные данные нарушают ограничения, то Администратор получает сообщение об ошибке а данные не сохраняются.

2.1.12 Блокирование работы биржи

1. Администратор биржи получает статус блокировки биржи
2. Администратор выставляет срочную блокировку биржи (указывая дату и время когда блокировка будет снята на любой момент в будущем).
3. Администратор сохраняет изменения

Алт.2 Администратор выставляет бессрочную блокировку биржи (выставляя дату и время снятия блокировки в 0).

Алт.2 Если ранее имелась блокировка, выставленная Администратором, то он может её снять (выставляя дату и время снятия блокировки в любое значение, предшествующее текущему моменту, но не 0).

2.2 Трейдер

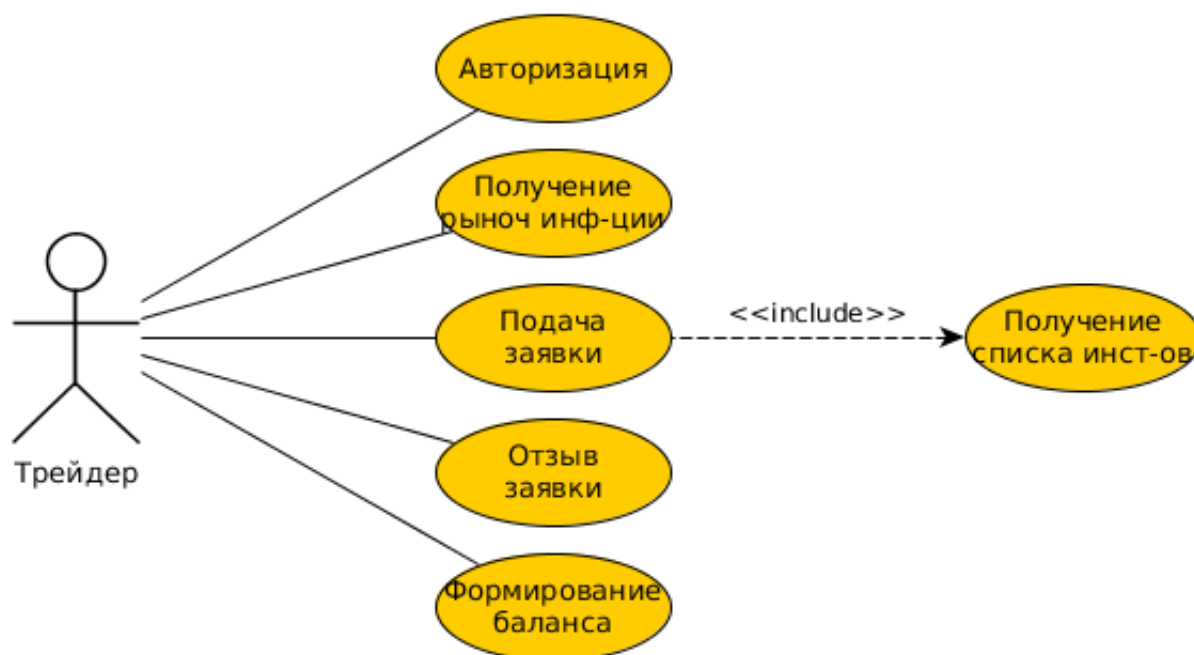


Рис. 2: Диаграммах прецедентов трейдера

2.2.1 Авторизация в программе

1. Для того чтобы выполнить вход в систему под пользователем Трейдер, необходимо ввести логин и пароль в специальной форме при запуске программы.

Алт.1 Если Пользователь не может выполнить авторизацию как Трейдер, он может продолжить работу как Анонимный пользователь.

Алт.1 Если Трейдер заблокирован (срочно или бессрочно) Администратором или Регулятором, то он не может завершить регистрацию, но может продолжить работу как Анонимный пользователь.

2.2.2 Получение статуса работы биржи

1. Трейдер отправляет запрос о статусе биржи, и получает статус готовности к работе

Алт.1 Если биржа заблокирована (срочно или бессрочно) Администратором или Регулятором, то будет получен статус о неготовности к работе

2.2.3 Получение списка торгуемых инструментов

1. Трейдер получает статус работы биржи
2. Трейдер отправляет запрос о получении списка торгуемых инструментов
3. Трейдер получает списка торгуемых (т.е. не заблокированных ни администрацией ни регулятором) инструментов

2.2.4 Получение рыночной информации

1. Трейдер получает статус работы биржи
2. Трейдер отправляет запрос о получении рыночной информации
3. Трейдер получает рыночную информацию с начала торговой сессии до текущего момента
4. Трейдер продолжает получать информацию об обновлении рыночной информации

2.2.5 Подача заявки на покупку/продажу инструмента

1. Трейдер получает статус работы биржи
2. Трейдер получает список торгуемых инструментов
3. Трейдер заполняет заявку и отправляет её на биржу
 - (Обязательно) выбрать ценную бумагу
 - (Обязательно) направление заявки (покупка/продажа)
 - (Обязательно) цена в копейках за 1 бумагу (больше 0)
 - (Обязательно) объём в ценных бумагах, согласно гранулярности биржи (больше 0)
 - (Опционально) время исполнения (больше текущего момента)

Алт.3 Если любое обязательное поле не заполнено или гранулярность не совпадает, Трейдер получит сообщение об ошибке, а заявка не будет принята.

2.2.6 Отзыв заявки на покупку/продажу инструмента

1. Трейдер получает статус работы биржи

2. Трейдер получает список своих активных (ещё не исполненных и не отозванных) заявок
3. Трейдер отправляет на биржу заявку с отзывом любой из своих активных заявок

2.2.7 Получение торгового баланса

1. Трейдер получает статус работы биржи
2. Трейдер отправляет на получение торгового баланса с начала торговой сессии
3. Трейдер получает список исполненных заявок и их стоимость

2.3 Анонимный пользователь

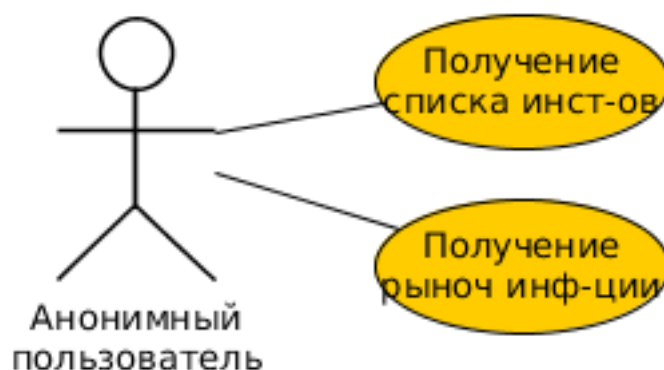


Рис. 3: Диаграмма прецедентов анонимного пользователя

2.3.1 Получение статуса работы биржи

1. Анонимный пользователь отправляет запрос о статусе биржи, и получает статус готовности к работе

Алт.1 Если биржа блокирована (срочно или бессрочно) Администратором или Регулятором, то будет получен статус о неготовности к работе

2.3.2 Получение списка торгуемых инструментов

1. Анонимный пользователь получает статус работы биржи

2. Анонимный пользователь отправляет запрос о получении списка торгуемых инструментов
3. Анонимный пользователь получает списка торгуемых (т.е. не заблокированных ни администрацией ни регулятором) инструментов

2.3.3 Получение рыночной информации

1. Анонимный пользователь получает статус работы биржи
2. Анонимный пользователь отправляет запрос о получении рыночной информации
3. Анонимный пользователь получает рыночную информацию с начала торговой сессии до текущего момента
4. Анонимный пользователь продолжает получать информацию об обновлении рыночной информации

2.4 Регулятор

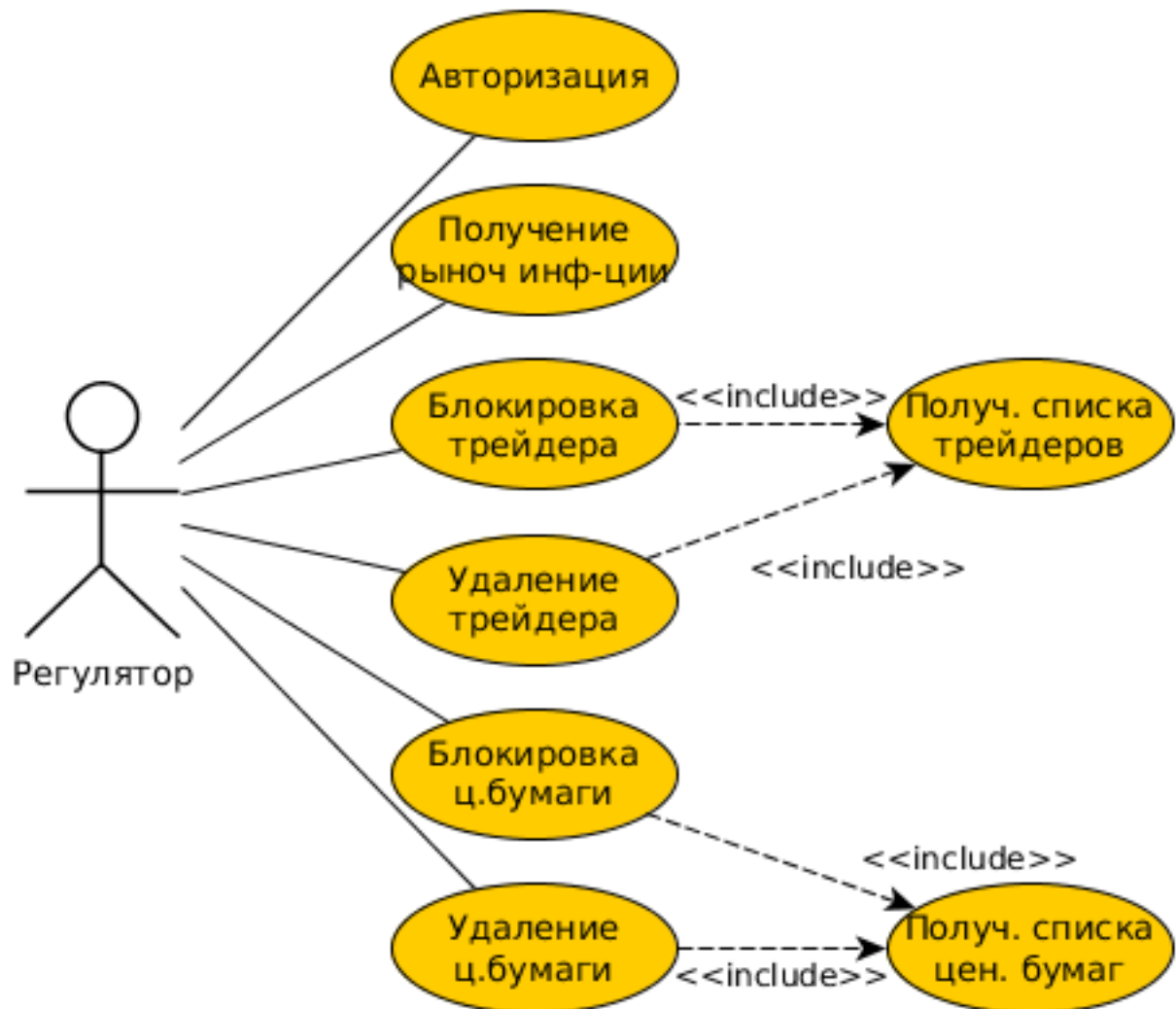


Рис. 4: Диаграммах прецедентов регулятора

2.4.1 Авторизация в программе

1. Для того чтобы выполнить вход в систему под пользователем Регулятор, необходимо ввести логин и пароль.

Алт.1 Если Пользователь не может выполнить авторизацию как Регулятор, он может продолжить работу как Анонимный пользователь.

2.4.2 Получение статуса работы биржи

1. Регулятор отправляет запрос о статусе биржи, и получает статус готовности к работе

Алт.1 Если биржа блокирована (срочно или бессрочно) Администратором или Регулятором, то будет получен статус о неготовности к работе

2.4.3 Получение рыночной информации

1. Регулятор получает статус работы биржи
2. Регулятор отправляет запрос о получении рыночной информации
3. Регулятор получает рыночную информацию с начала торговой сессии до текущего момента
4. Регулятор продолжает получать информацию об обновлении рыночной информации

2.4.4 Блокирование работы биржи

1. Регулятор биржи получает статус блокировки биржи
2. Регулятор выставляет срочную блокировку биржи (указывая дату и время когда блокировка будет снята на любой момент в будущем).
3. Регулятор сохраняет изменения

Алт.2 Регулятор выставляет бессрочную блокировку биржи (выставляя дату и время снятия блокировки в 0).

Алт.2 Если ранее имелась блокировка, выставленная Регулятором, то он может её снять (выставляя дату и время снятия блокировки в любое значение, предшествующее текущему моменту, но не 0).

2.4.5 Блокирование трейдера

1. Регулятор биржи получает статус блокировки биржи
2. Регулятор биржи получает список трейдеров
3. Регулятор выставляет срочную блокировку трейдера (указывая дату и время когда блокировка будет снята на любой момент в будущем).
4. Регулятор сохраняет изменения

Алт.3 Регулятор выставляет бессрочную блокировку трейдера (выставляя дату и время снятия блокировки в 0).

Алт.3 Если ранее имелась блокировка, выставленная Регулятором, то он может её снять (выставляя дату и время снятия блокировки в любое значение, предшествующее текущему моменту, но не 0).

2.4.6 Блокирование ценной бумаги

1. Регулятор биржи получает статус блокировки биржи
2. Регулятор биржи получает список ценных бумаг
3. Регулятор выставляет срочную блокировку ценной бумаги (указывая дату и время когда блокировка будет снята на любой момент в будущем).
4. Регулятор сохраняет изменения

Алт.3 Регулятор выставляет бессрочную блокировку ценной бумаги (выставляя дату и время снятия блокировки в 0).

Алт.3 Если ранее имелась блокировка, выставленная Регулятором, то он может её снять (выставляя дату и время снятия блокировки в любое значение, предшествующее текущему моменту, но не 0).

3 Статическая модель предметной области

Статическая модель предметной области представлена диаграммой классов (Static Structure Diagram).

Существуют разные точки зрения на построение диаграмм классов в зависимости от целей их применения:

- Концептуальная точка зрения – диаграмма классов описывает модель предметной области, в ней присутствуют только классы прикладных объектов;
- Точка зрения спецификации – диаграмма классов применяется при проектировании информационных систем;
- Точка зрения реализации – диаграмма классов содержит классы, используемые непосредственно в программном коде (при использовании объектно-ориентированных языков программирования).

Реализацию можно автоматически получить из исходного кода. На рисунке 5 представлена реализация слоя сервисов, на рисунке 6 – слоя инфраструктуры. Отображение более сложных слоёв или сразу нескольких слоёв на одной диаграмме затруднительно для понимания.

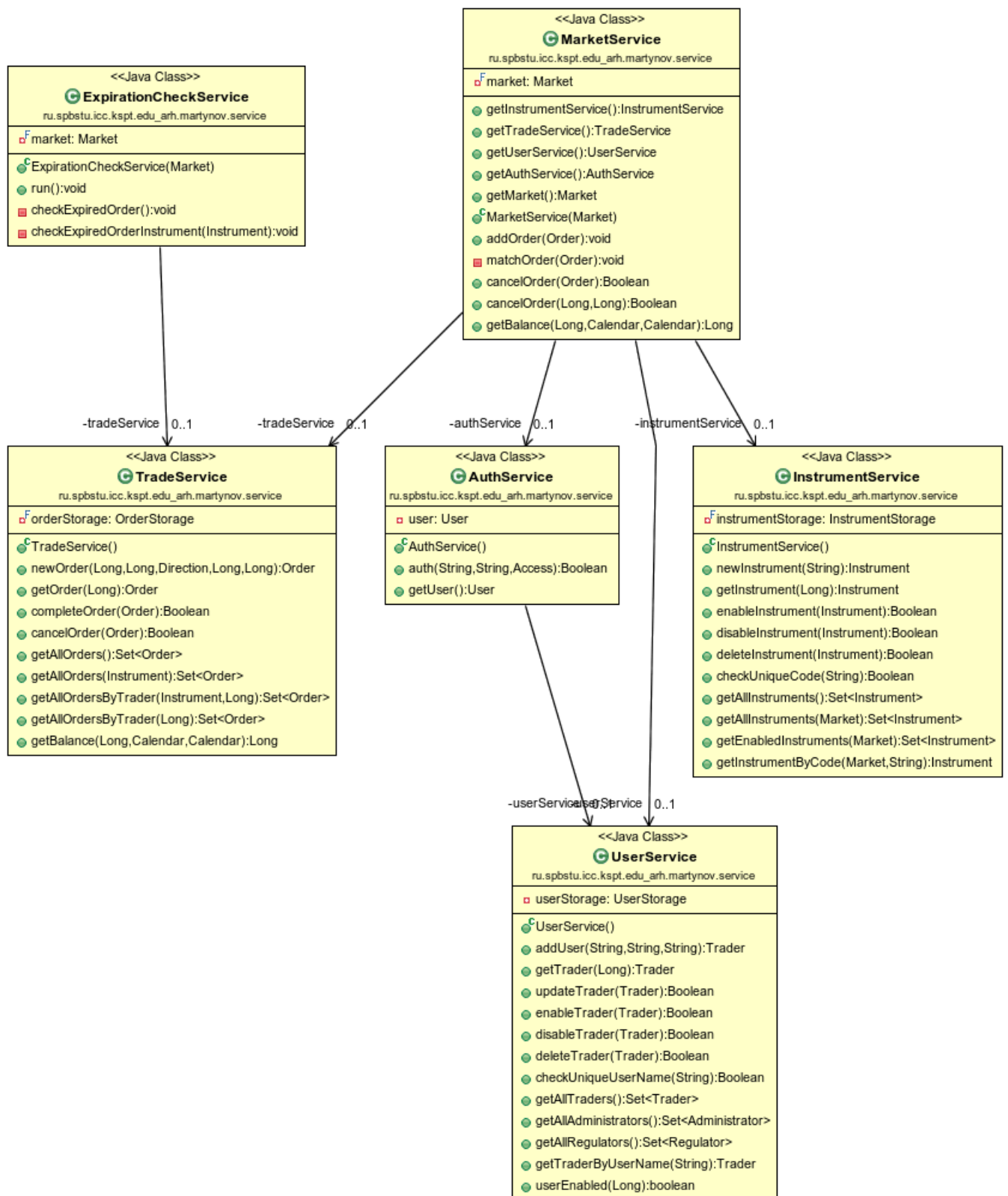


Рис. 5: Слой сервисов

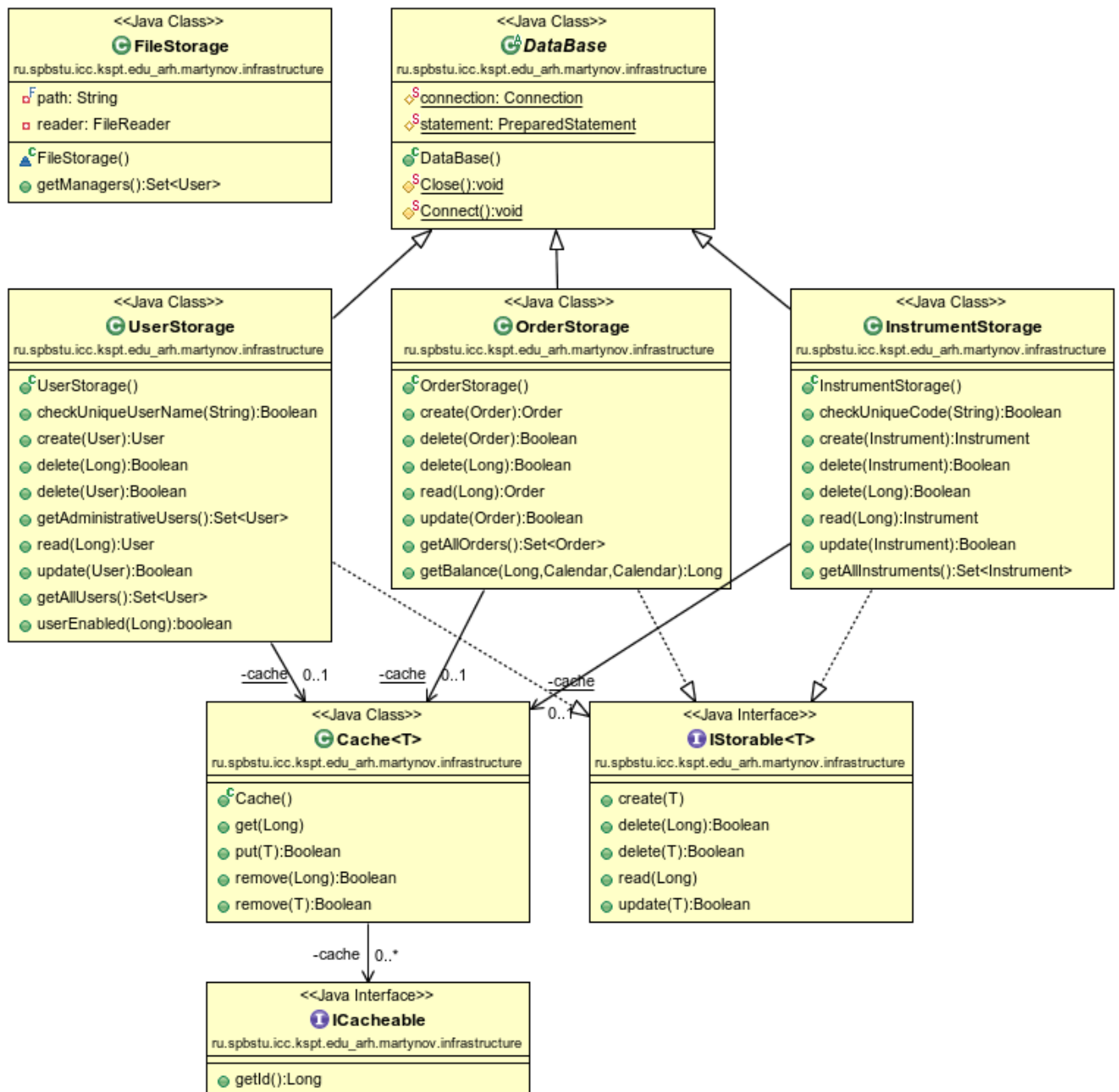


Рис. 6: Слой инфраструктуры

Более общая диаграмма представлена на рисунке 7.

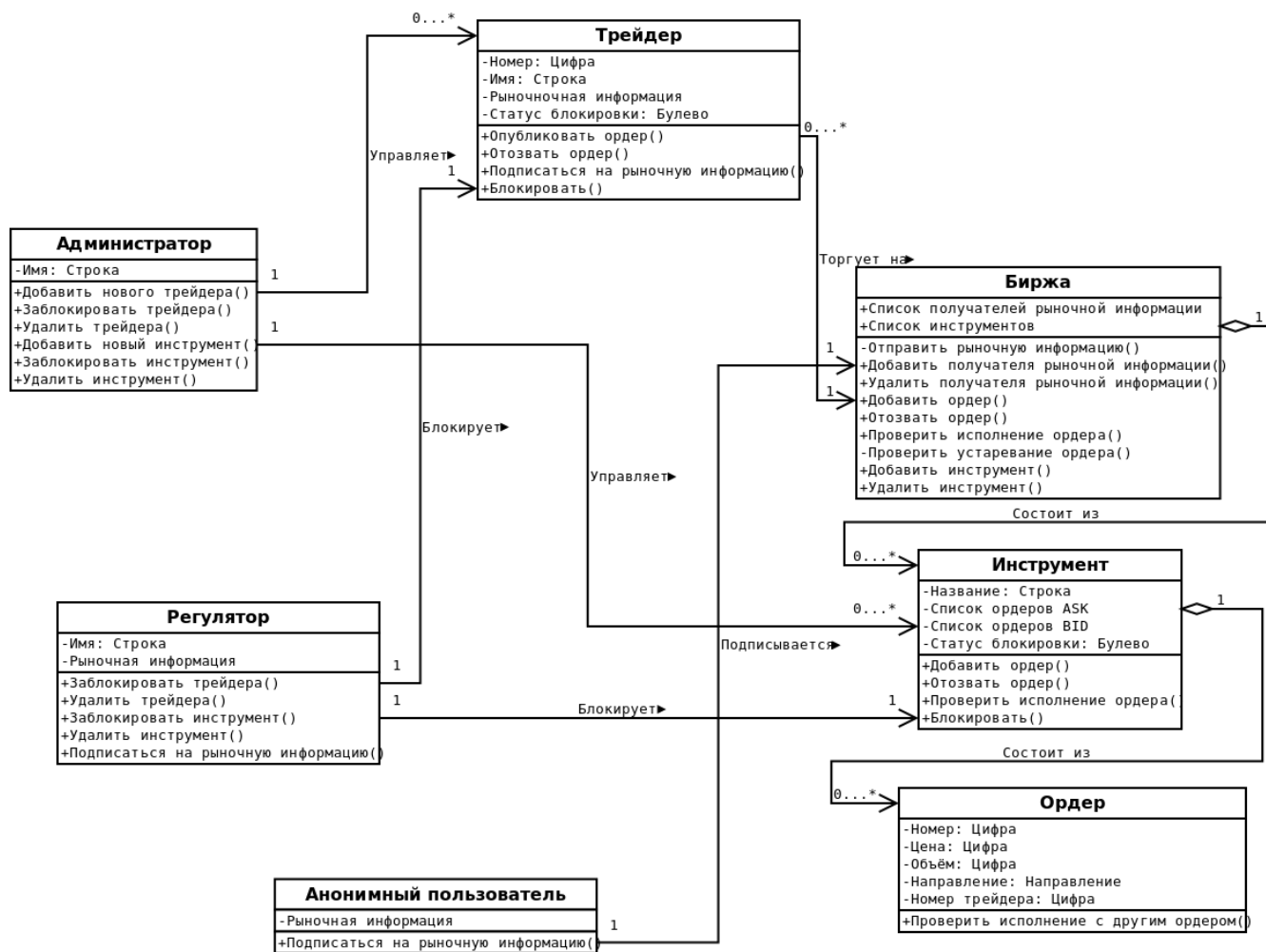


Рис. 7: Диаграмма классов

Анонимный пользователь, администратор, регулятор и трейдер имеют общий функционал – подписка на рыночную информацию.

Рыночная информация представлена отдельным классом, и состоит из записей MarketRecord. Формат представления соответствует Московской бирже и описывает три события: появление новой заявки, удаление заявки и её выполнение (запись о выполнении будет повторяться два раза т.к. в ней участвуют две заявки).

Трейдер генерирует заявки и передаёт их бирже. Биржа раскладывает заявки по инструментам, где они хранятся в виде стакана – отдельно ask, отдельно bid.

Все исполняемые заявки проходят через класс, определяющий правила торговли. На этом этапе вычисляется доля биржи.

4 Динамическая модель предметной области

Динамическая модель предметной области представлена диаграммой последовательности (Sequence Diagram).

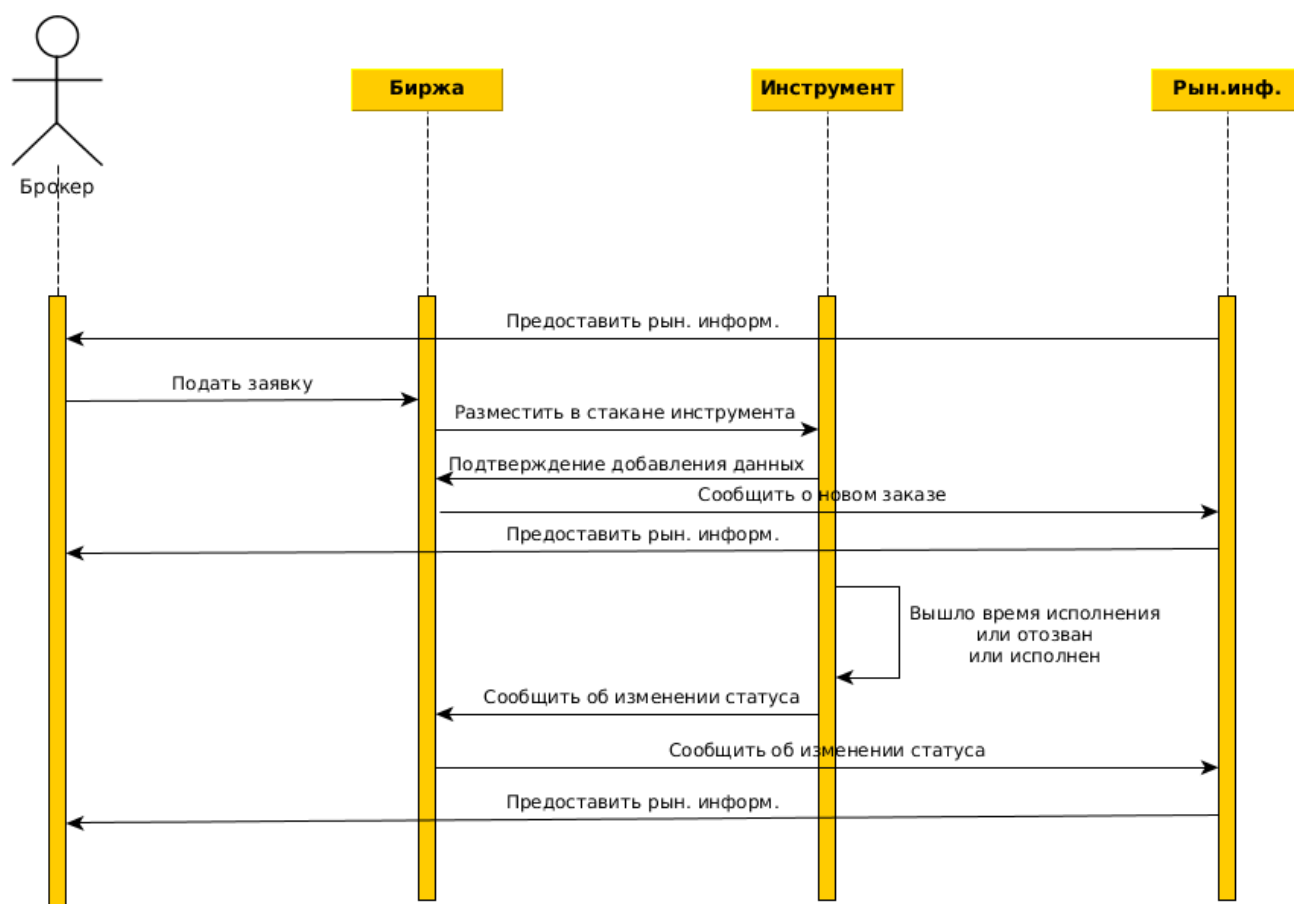


Рис. 8: диаграммой последовательности заказа

Когда брокер формирует новую заявку на покупку или продажу, он передаёт эту заявку на биржу. Биржа хранит заявки по инструментам в стаканах, в зависимости от направления сделки. Как только инструмент принял заявку, информация об этом отображается в Рыночной информации, что могут видеть все участники торгов.

Заявка находится в Инструменте до тех пор, пока её не отзовёт сам брокер, либо не закончится её время исполнения, либо она не будет исполнена.

В любом случае, изменение состояния заявки сразу отображается в рыночной информации (с соответствующим кодом) и об этом сообщается трейдеру, разместившему заявку.

5 Архитектура приложения

В процессе работы, архитектура расслоилась на 5 слоёв, взаимодействие которых показано на рисунке 7.

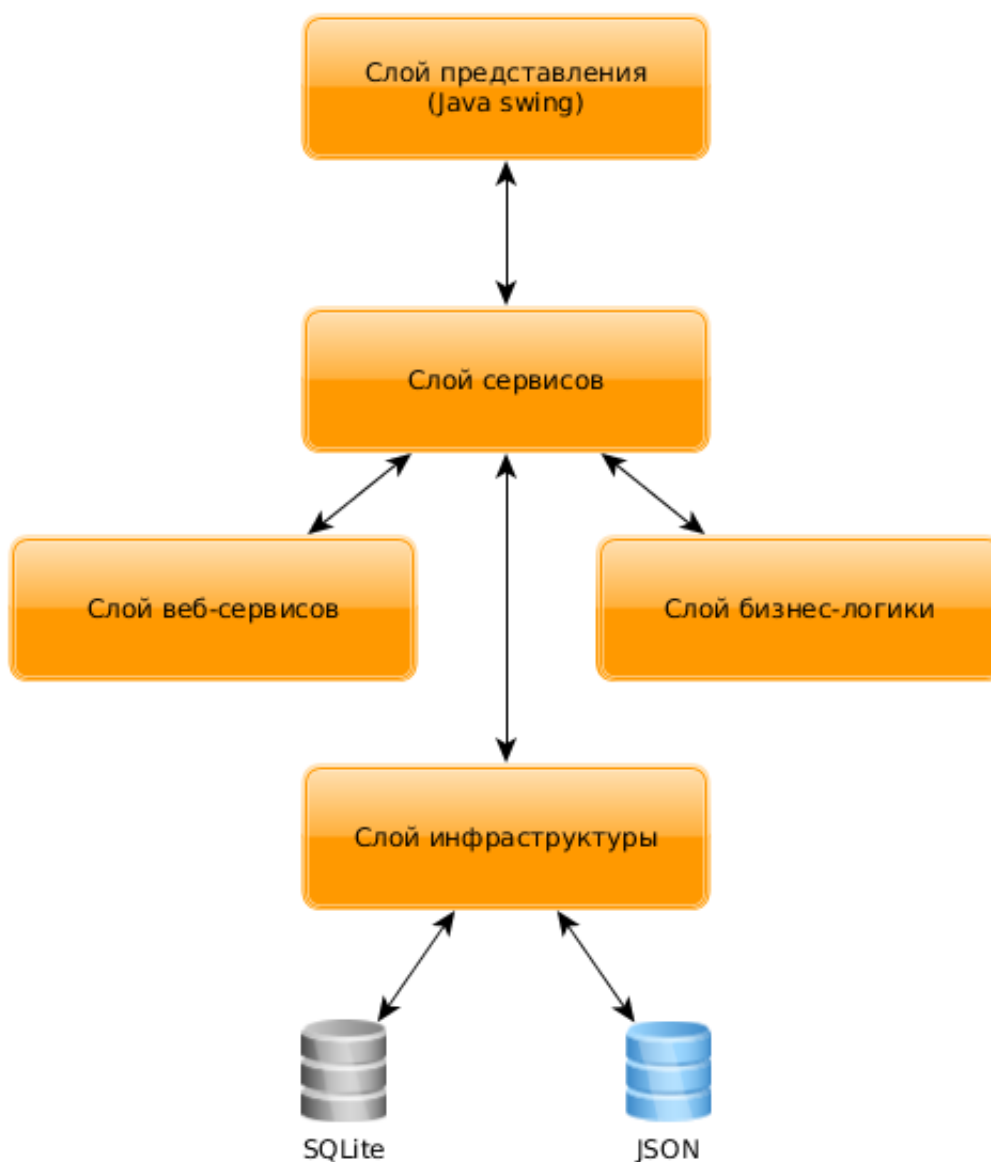


Рис. 9: Взаимодействие слоёв архитектуры

5.1 Бизнес логика

В процессе выбора решения по организации бизнес логики были рассмотрены следующие типовые решения:

- Сценарий транзакции (Transaction Script)

- Модель предметной области (Domain Model)
- Модуль таблицы (Table Module)

Как подчёркивается в [1], рассмотренные решения не являются взаимоисключающими альтернативами и нередко используются совместно для различных частей бизнес логики. Развивая эту идею, работа над реализацией бизнес логики была разделена на два слабо связанных сегмента: один отвечал за всё конфигурирование биржи, другой за торги.

Конфигурирование биржи фактически сводится к простому сохранению и чтению значений в/из хранилища (базы данных). Тут отсутствует какая-то сложная логика, кроме проверки данных на ошибки. Для решения подобных случаев отлично подходит решение Сценарий транзакции.

Торги, напротив, предполагают наличие относительно сложной но однообразной логики, которая затрагивает одновременно большое количество таблиц в базе данных. Кроме того, предметная область требует высокой скорости работы биржи, потому все не завершённые заявки должны постоянно находиться в памяти, что делает этот ресурс сильно востребованным. В этой ситуации наиболее подходящим решением представляется Модуль таблицы.

Между тем, практическая реализация рассмотренных подходов сильно уступала решению Модели предметной области, по крайней мере по части скорости разработки. В [1] такая ситуация объясняется предыдущим опытом, который имеет разработчик информационной системы. Учитывая, что все типовые варианты взаимозаменяемы, окончательное решение было в пользу Модели предметной области, в той её разновидности, которую сам Фаулер называет "простой т.е. структура во многом напоминает структуру базы данных.

5.2 Инфраструктурный слой

С одной стороны, источником данных в информационной системе является пользователь (трейдер), с другой – имеется необходимость постоянного хранения рыночной информации и настроек.

В качестве типового решения архитектурны источника данных были рассмотрены следующие:

- Шлюз таблицы данных (Table Data Gateway)
- Шлюз записи данных (Row Data Gateway)
- Активная запись (Active Record)

- Преобразователь данных (Data Mapper)

Фаулер в [1] приводит таблицу соответствия типовых решений инфраструктурного слоя решениям бизнес-логики. На предыдущем шаге нами было выбрано решение Модель предметной области, таким образом выбор сводился к Активной записи или Преобразователю данных. Шлюз таблицы данных допустимых хоть и допустим, но больше подходит для Модуля таблицы т.к. предоставляет возможности удобной работы с множеством выборок из базы данных (RecordSet). А Шлюз записи данных удобнее всего использовать со Сценарием транзакции т.к. обеспечивает практически сквозную работу с БД, поля объекта есть поля кортежа выборки. Но для выбранной нами Модели предметной области это подходит плохо т.к. сильно увеличивается потребление памяти из-за разрастания представления объектов.

Между Активной записью и Преобразователем данных предпочтение было отдано предпочтению второму. Активная запись хороша, когда к данным хочется добавить логику домена, но в нашем случае вся логика собрана в слое бизнес-логики и сервисном слое. В то же время, было желание максимальной изоляции БД от приложения для лёгкой замены сервера БД – реляционные базы данных, колоночные, или просто XML-файлы...

5.3 Сервисный слой

Были рассмотрены следующие варианты реализации сервисного слоя:

- Интерфейс доступа к домену (Domain Facade)
- Сценарий операции (Operation Script)

Изначально была выполнена реализация более простого Интерфейса доступа к домену (фактически слой занимался тем, что отдавал рыночные данные и принимал внешние запросы), но это привело к большой сложности в слое бизнес-логики. После этого было принято решение переписать этот слой используя Сценарий операций для разгрузки бизнес-логики.

5.4 Слой представления

Слой представления для администратора написан на фреймворке Swing. Он позволяет выполнять базовые операции, согласно функциональным требованиям.

Треjder тоже имеет свой графический интерфейс, написанный с использованием фреймворка Swing, но он имеет минимальную функциональность, и, как правило, имея описание

работы протокола многие разработчики предпочитают делать свою реализацию, снабжая её большими аналитическими функциями.

На рисунке 10 представлено окно входа в систему, на рисунке 11 и 12 интерфейс трейдера и регулятора.

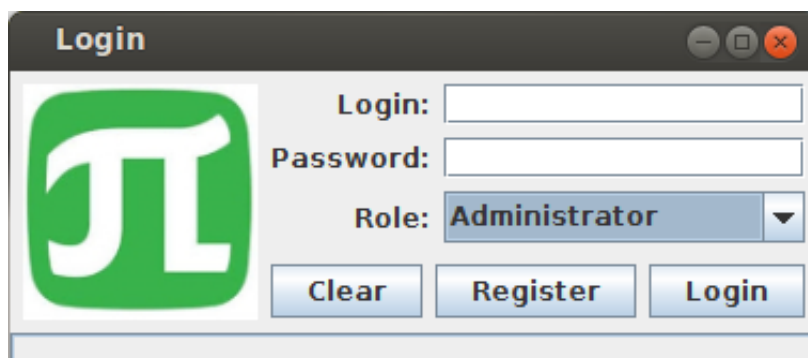


Рис. 10: Окнов входа в систему

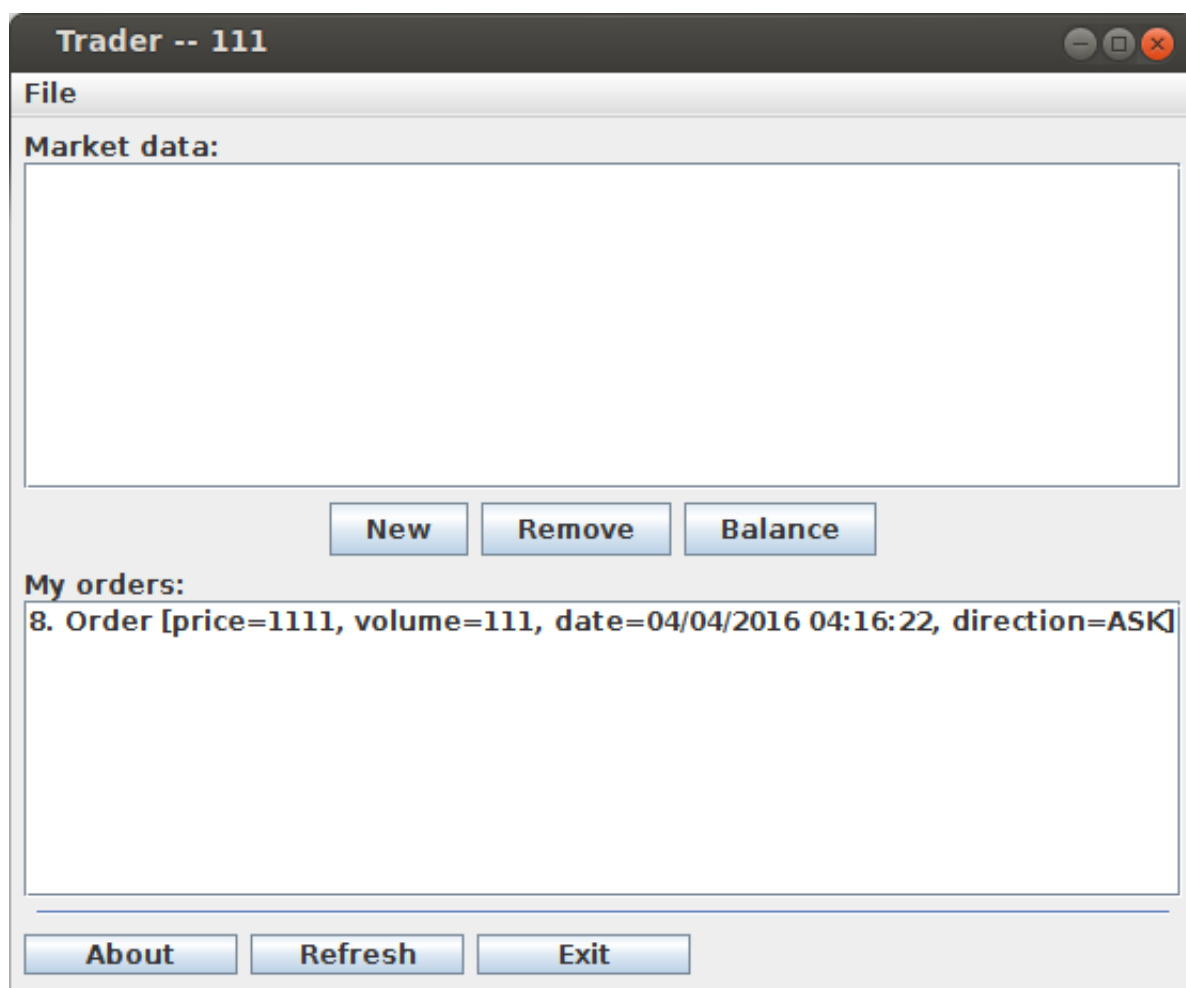


Рис. 11: Интерфейс трейдера

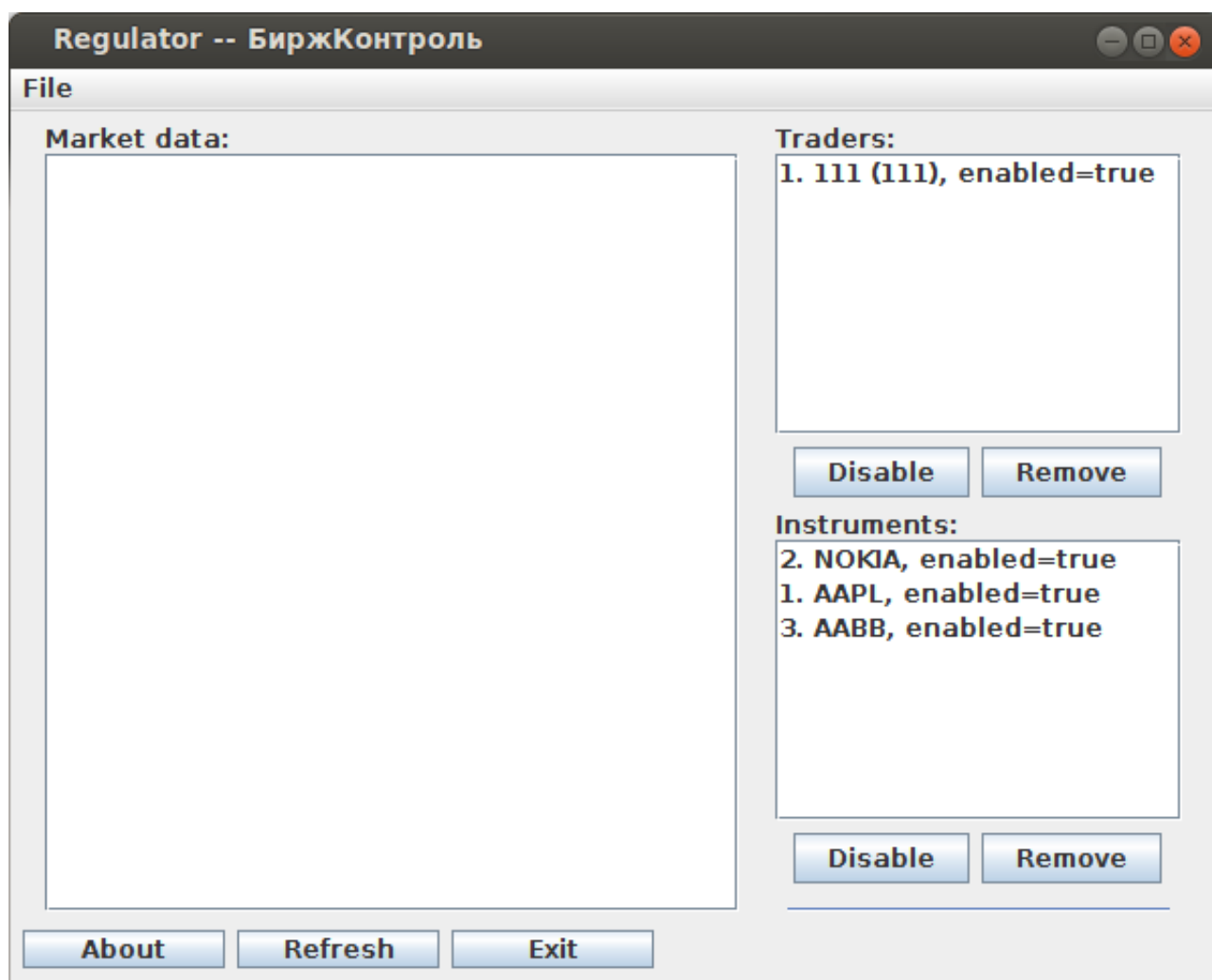


Рис. 12: Интерфейс регулятора

Анонимный пользователь имеют только API. Обычно за ними стоят различные сервисы анализа данных.

5.5 Слой веб-сервисов

API для удалённых клиентов реализовано по средствам простых запросов, релизованных поверх HTTP. Текстовые протоколы не могут обеспечить быстродействие, требуемое в данной предметной области, но они просты и надёжны.

6 Тестирование

Модульное тестирование, или юнит-тестирование (Unit testing) – методология тестирования, при которой определенные модули программы тестируются отдельно от остальных модулей. Для выполнения юнит-тестирования в Java наибольшей популярностью пользуется фреймворк под названием JUnit4, который мы использовали для тестирования в процессе работы.

JUnit является продуктом с открытым исходным кодом, и достаточно просто интегрируется со средой разработки Eclipse.

В данной работе мы очень поверхностно использовали возможности данного фреймворка, хотя на реальных проектах он позволяет тестировать большой объём кода и экономить время разработчиков.

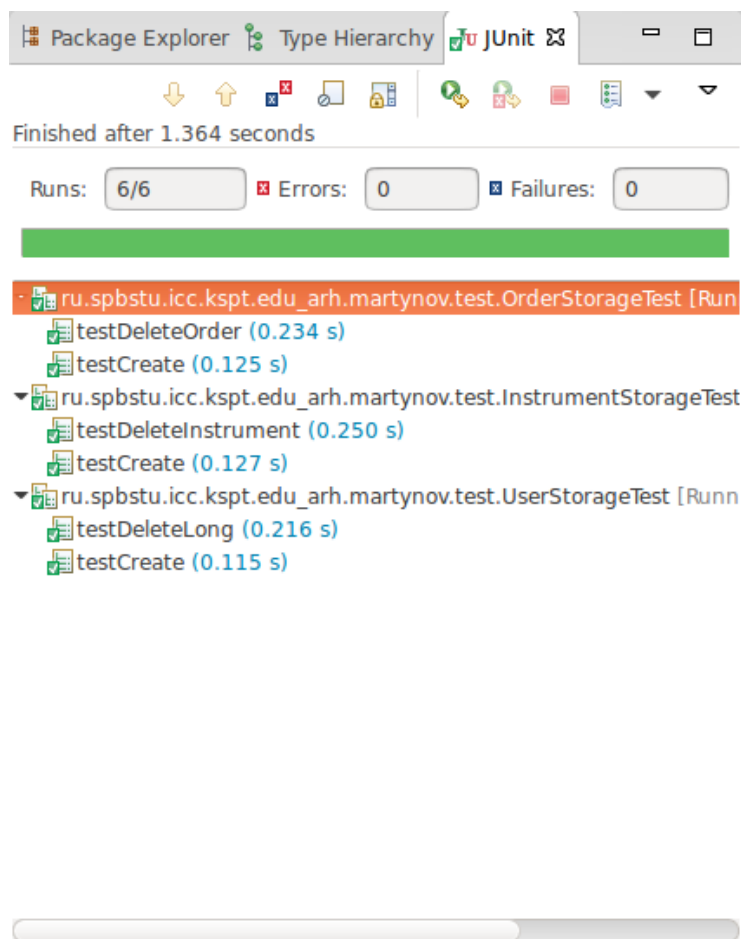


Рис. 13: Результаты тестирования

Заключение

В результате проектирования системы были изучены различные паттерны проектирование, проведен анализ их возможного использования и в итоге часть из них применена для написания курсовой работы. Были получены навыки проектирования системы от сбора функциональных требований до реализации готового работоспособного приложения, выполняющего требуемые функции. Можно добавить, что знание паттернов проектирования для разработчика является ключевой вещью, т.к. помогает грамотно разрабатывать проекты, повышая эффективность, так же в дальнейшем такие проекты легки для сопровождения.

Список литературы

- [1] Архитектура корпоративных программных приложений / Фаулер М, Райс Д., Фоммел М. и др. – Вильямс, 2007 – 544 с.
- [2] Приемы объектно-ориентированного проектирования. Паттерны проектирования / Гамма Э., Хелм Р., Джонсон Р. и др. – Питер, 2000 – 366 с.