**Функциональные требования**

**В информационной системе предусмотренны следующие роли**

* Букмекер - представитель букмекерской конторы, осуществляет контроль за компаниями и пользователями
* Компания - представитель компаний создающих события и исходы на которые производятся ставки
* Пользователь - непосредственно участник ставок. Выбирает события и делает ставки на определенные исходы

**Функциональные требования**

1) Для пользователя:

* Пополнение средств
* Отображение доступных событий и исходов
* Создание ставки
* Пополнение средств
* Пополнение счета при успешном исходе ставки
* Снятие средств

2) Для компании:

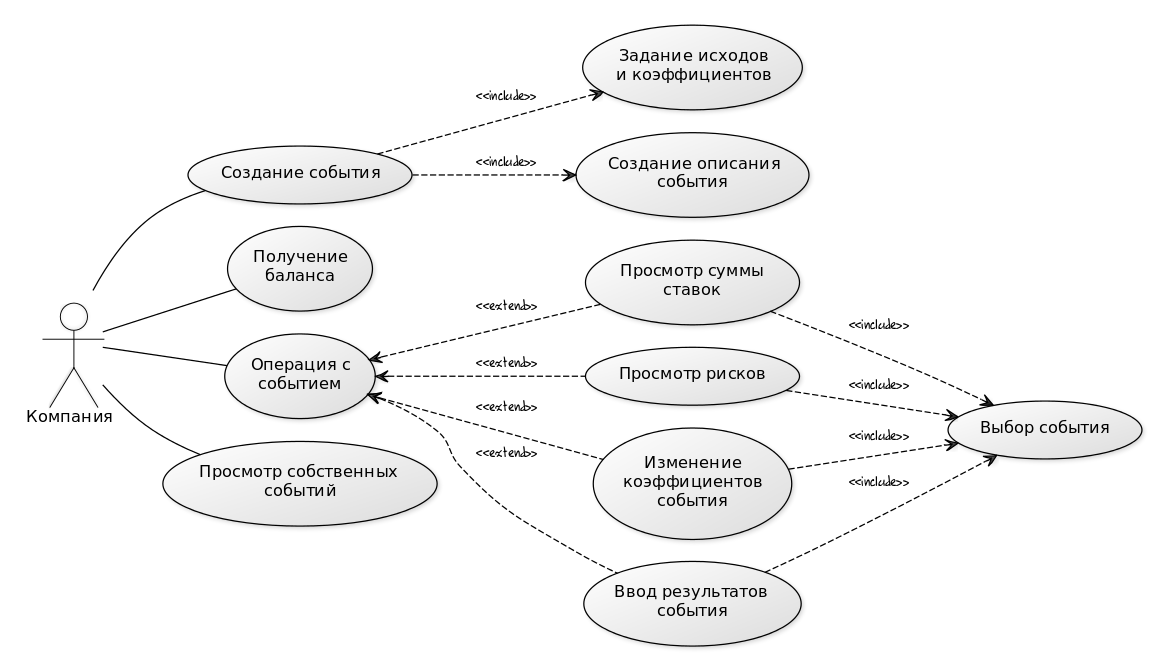
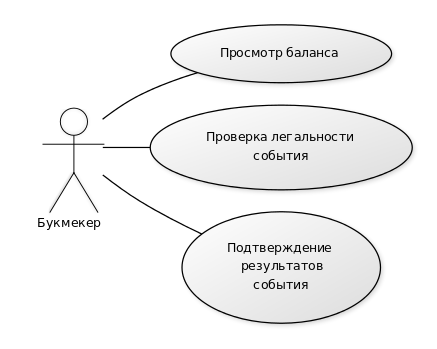
* Создания события с исходами
* Изменение коэффициентов события
* Отображение рисков
* Отображение общей суммы ставок
* Перерасчет прибыли при определении результатов исхода
* Просмотр баланса

3) Для букмекера:

* Подтверждение результатов события
* Проверка легальности события
* Просмотр баланса

**Разработка вариантов использования**

Обобщенная диаграмма прецедентов:



Подробное описание вариантов использования

### 1. Пользователь

**1.1. Пополнение средств**

1. Пользователь авторизуется
2. Система предоставляет ему реквизиты
3. Пользователь вносит средства используя иные инструменты

**1.2. Просмотр баланса**

1. Пользователь авторизуется
2. Система предоставляет информацию о балансе

**1.3. Снятие средств**

1. Пользователь авторизуется
2. Пользователь заказывает вывод средств на сторонние счета
3. Система проверяет наличие указанной суммы

Альтернатива: указанной суммы нет

Заявка отменяется

1. Средства списываются со счета
2. Пользователь ожидает поступления на стороннем счете

**1.4. Просмотр доступных событий**

1. Пользователь авторизуется
2. Система предоставляет информацию о доступных событиях, их исходов и коэф.

**1.5. Создание ставки**

1. Пользователь авторизуется
2. Система предоставляет информацию о доступных событиях, их исходов и коэф.
3. Пользователь выбирает одно из событий
4. Пользователь указывает исход и сумму ставки
5. Пользователь подтверждает создание ставки
6. Система проверяет наличие указанной суммы

Альтернатива: указанной суммы нет

Операция отменяется

1. Система регистрирует указанную ставку

### 2. Компания

**2.1. Получение баланса**

1. Представитель компании авторизуется
2. Система предоставляет информацию о балансе

**2.2. Создание события**

1. Представитель компании авторизуется
2. Представитель вводит описание события
3. Представитель вводит возможные исходы и коэффициенты
4. Представитель подтверждает создание нового исхода

**2.3. Просмотр собственных событий**

1. Представитель компании авторизуется
2. Система предоставляет информацию о созданных событиях

**2.4. Операция над событием**

1. Представитель компании авторизуется
2. Представитель просматривает список событий компании и выбирает одно из них

**2.4.1 Просмотр суммы ставок**

1. Выполняются действия из 2.4
2. Система предоставляет информацию о сумме ставок на каждый исход

**2.4.2 Просмотр рисков**

1. Выполняются действия из 2.4
2. Система предоставляет информацию о сумме рисков по каждому исходу

**2.4.3 Изменение коэффициентов**

1. Выполняются действия из 2.4
2. Представитель указывает новые коэффициенты для каждого исхода
3. Представитель подтверждает изменения

**2.4.4 Ввод результатов события**

1. Выполняются действия из 2.4
2. Представитель указывает победивший исход
3. Система блокирует событие для новых ставок и изменений
4. Событие ожидает подтверждение Букмекера (п 3.3)

### 3. Букмекер

**3.1. Просмотр баланса**

1. Представитель букмекерской конторы (букмекер) авторизуется
2. Система предоставляет информацию о текущем балансе

**3.2. Проверка легальности события**

1. Букмекер авторизуется
2. Букмекер выбирает событие
3. Букмекер запрашивает статус легальности события
4. Система осуществляет запрос к стороннему сервису и предоставление полученной информации

**3.3. Подтверждение результатов события**

1. Букмекер авторизуется
2. Букмекер выбирает событие
3. Букмекер подтверждает победивший исход

Альтернатива: букмекер не подтверждает правдивость данных и указывает другой исход

Система фиксирует указанный букмекером исход как победивший

1. Система выплачивает соответствующую сумму победившим Пользователям
2. Система начисляет комиссию на счет Букмекера
3. Система выплачивает Компании проигравшие ставки
4. Система меняет статус события на “выплачен”

**Разработка статической объектной модели предметной области**

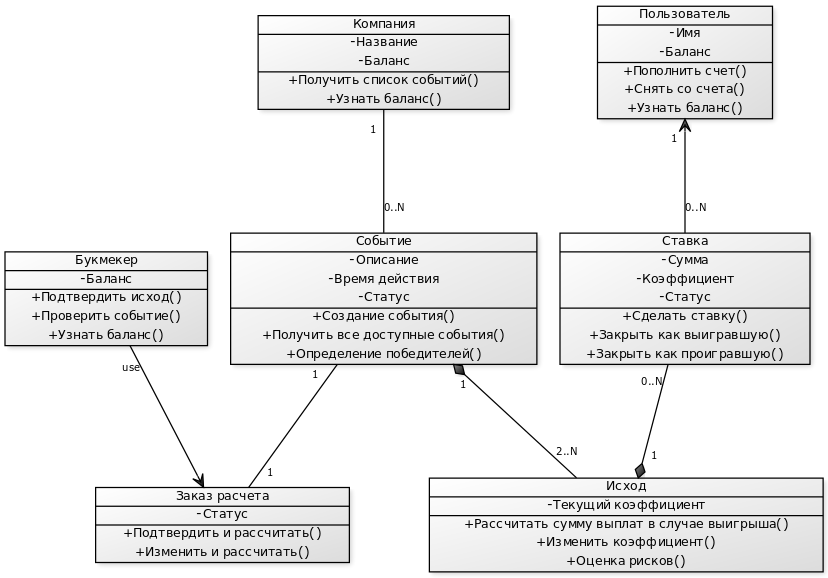


Рис. 2. Диаграмма классов.

**Разработка динамической объектной модели предметной области**

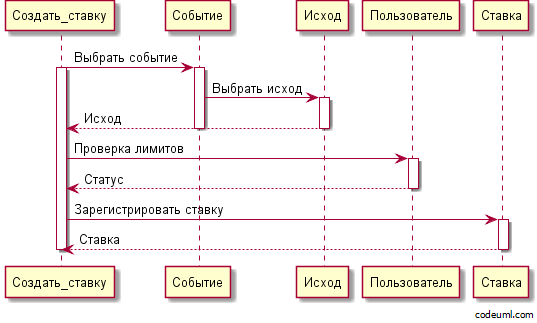
****

Рис. 3. Диаграмма последовательности «Создание ставки».

**Архитектурные шаблоны**

### Бизнес логика

Для реализации бизнес логики выбран шаблон "Domain model". Подход предполагает создание модели предметной области вместе с последующей работой в рамкой этой модели. Подход сложнее в реализации, однако предоставляет большую гибкость при модификации. Выгода метода особенно заметна для больших проектов.

Domain model и не является правильным решением для данного задания. Однако, так как это учебный проект, предпочтение было отдано более популярному на практике подходу.

### Слой доступа к данным

Слой доступа к данным реализован на основе Active record. Данный подход предполагает работу с записями таблицы, создавая при этом для каждой записи новый объект. Такой прием является простым в реализации и хорошо сочетается с шаблоном Domain model. В качестве источника данных используется база данных MySql.

### Сервисный слой

Как и в предыдущем случае, выбор паттерна для бизнес логики во многом определил выбор паттерна для сервисного слоя. В данном случае это Domain Facade. Так как в задании не предполагается клиентов, использующих платформу отличную от java, для взаимодействия была выбрана технология RMI.

### Слой представления

Представление реализовано при помощи java swing. Swing предполагает собственные концепции разработки графического интерфейса и собственную архитектуру.

**Реализация**

В качестве базы данных используется MySql. Структура базы данных представлена на рис. 4.

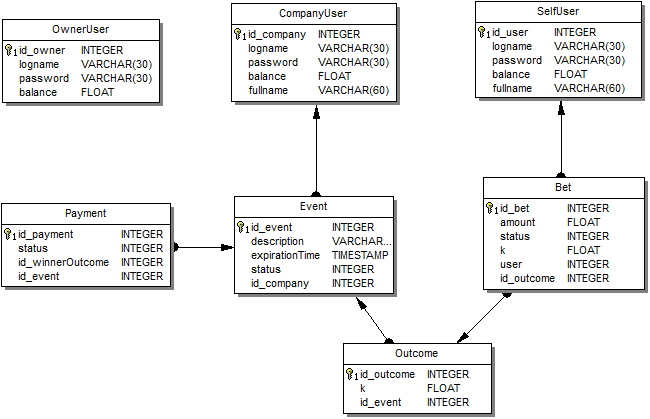


Рис. 4. Структура БД.

Для реализации слоев бизнес логики, доступа к данным и сервисного слоя в проекте BetService созданы соответствующие пакеты. Сервисный слой реализован при помощи RMI и предоставляет три типа интерфейсов: для пользователя, для компании и для владельца площадки

GUI реализован в виде отдельного проекта «GUI» и предоставляет доступ к трем типам интерфейсов: для пользователя, для компании и для владельца площадки. Также GUI включает в себя интерфейсы сервисов, для удаленной связи с ними.

**Тестирование**

Кроме сервисного пакета, для всех пакетов написаны Unit тесты. На рис. 5. можно увидеть текущее покрытие.

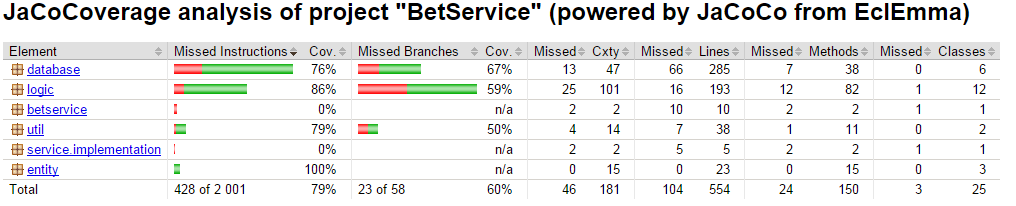


Рис. 5. Покрытие кода тестами.

**Приложение**

Репозиторий проекта с исходным кодом, дампом базы данных и другими используемыми материалами находится на https://github.com/suhininalex/Bet