

ГЛАВА 4. ПРОЕКТИРОВАНИЕ КЛАССОВ

Данная глава посвящена выделению объектов системы, на основе которых далее описываются классы и связи между ними, уточняются атрибуты классов, формируется сначала диаграмма концептуальных классов, а далее диаграмма взаимодействий, состояний и диаграмма проектных классов.

4.1 Диаграмма концептуальных классов

В результате анализа предметной области (работа тату-мастера, взаимодействие с клиентами, оформление заявок через сайт, хранение портфолио, информации об услугах и сопутствующих данных) была выделена следующая концептуальная модель. Она отражает все основные объекты системы, их свойства и связи, что позволяет организовать базу данных и кодовое представление системы.

В модели выделено несколько типов сущностей:

- клиент (Client) — хранит персональные данные посетителя, которые необходимы для связи (имя, контакт, возраст)
- заявка (UserRequest) — представляет факт отправки клиентом запроса: ссылается на клиента, содержит дату заявки, её статус
- тату-идея (TattooIdea) — хранит детали желаемой татуировки: часть тела, цвет/цветность, размер, дополнительные комментарии, желаемое время записи и др.
- референс-файл (ReferenceFile) — изображения/файлы, прикрепленные клиентом как референсы к тату-идее
- проверка возраста (AgeVerification) — сущность, фиксирующая факт верификации возраста клиента (например, достиг ли 18 лет), для возможности дальнейшего оформления заявки

Связи между сущностями отражают бизнес-логику: клиент может отправлять множество заявок; каждая заявка содержит одну тату-идею; к одной тату-идее может быть прикреплено несколько референс-файлов; справочные сущности (портфолио, цены, информационные разделы) существуют независимо от заявок; мастер получает заявки; проверка возраста связывается с клиентом.

Такая структура покрывает все основные требования системы: сбор заявки, хранение информации о клиентах и их пожеланиях, возможность расширения — добавления новых работ в портфолио, изменения цен, управление информацией и т.п.

№	Класс	Основание для выделения / значение в системе
1	Client (Клиент)	Клиент вводит свои персональные данные в форме заявки; необходим для обратной связи, проверки возраста и идентификации отправителя.
2	AgeVerification (Проверка возраста)	Требование системы — проверять, достиг ли пользователь 18 лет; влияет на возможность оформления заявки и безопасность услуг.
3	UserRequest (Заявка)	Центральный объект бизнес-процесса: фиксирует отправку формы, привязывает клиента к идее татуировки, хранит статус заявки.
4	TattooIdea (Идея татуировки)	Хранит содержательную часть заявки — параметры будущей татуировки (размер, часть тела, цветность, комментарии), что является ключевым объектом предметной области.
5	ReferenceFile (Референс-файл)	Клиент может прикреплять изображения-примеры. Файлы должны храниться как отдельные сущности, связанные с идеей татуировки.

На следующем этапе проектирования определяются связи-ассоциации между выделенными классами.

№	Название связи	Выход (класс-источник)	Вход (класс-приёмник)	Категория	Назначение
1	Клиент отправляет заявку	Client	UserRequest	Ассоциация	Связывает клиента с отправленными им заявками. Один клиент может создать несколько заявок.

№	Название связи	Выход (класс-источник)	Вход (класс-приёмник)	Категория	Назначение
2	Клиент проходит проверку возраста	Client	AgeVerification	Ассоциация	Позволяет системе определить, достиг ли клиент 18 лет и может ли он продолжить оформление заявки.
3	Заявка содержит описание татуировки	UserRequest	TattooIdea	Композиция	Тату-идея является содержательной частью заявки и не существует отдельно от неё.
4	Идея татуировки включает прикрепленные файлы	TattooIdea	ReferenceFile	Композиция	Референсы принадлежат конкретной тату-идее и удаляются вместе с ней.

В концептуальную диаграмму включаются только те атрибуты, которые действительно необходимы для хранения важной информации.

Принципы выбора атрибутов: атрибут должен иметь простой тип (строка, число, дата) сложные типы оформляются через ассоциации на данном уровне поведение отражается минимально.

Ниже приведены примеры атрибутов выделенных классов:

- Client: clientID, name, email, phone
- UserRequest: requestID, submissionDate, status
- AgeVerification: verificationID, verifiedAt, isAdult
- TattooIdea: ideaID, description, style, placement
- ReferenceFile: fileID, fileName, fileType, uploadedAt

В классах UserRequest и TattooIdea были добавлены массивы объектов:

- UserRequest содержит массив объектов TattooIdea, так как одна заявка может включать несколько идей татуировки
- TattooIdea содержит массив объектов ReferenceFile, чтобы хранить все файлы, относящиеся к конкретной тату-идее.

Необходимость включения массивов объясняется тем, что оба класса по своей логике представляют коллекции связанных объектов. Добавление массивов позволяет корректно организовать связь «один-ко-многим» между заявкой/тату-

идеями и отдельными объектами, упрощает поиск, добавление и удаление элементов, а также обеспечивает удобную работу с данными при обработке заявок и управлении тату-идеями.

Такое решение делает систему более гибкой, структурированной и соответствующей требованиям объектно-ориентированного проектирования.

На основе представленных выше данных была разработана диаграмма концептуальных классов, включающая ключевые сущности, их атрибуты и ассоциации.

Данная диаграмма представлена на рисунке 4.1.

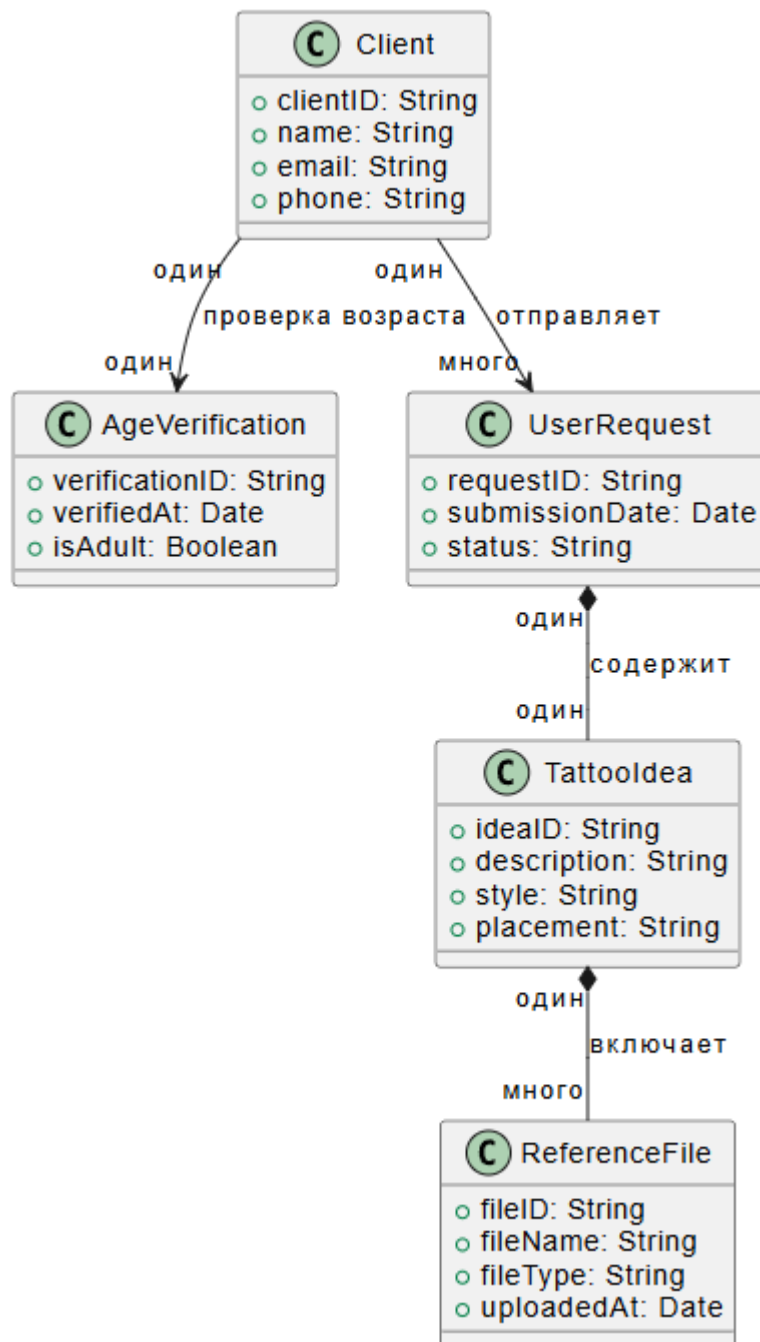


Рисунок 4.1 – Диаграмма концептуальных классов

Источник: собственная разработка

Диаграмма отражает ключевые сущности предметной области работы тату-салона, их основные атрибуты и взаимосвязи. На схеме представлены классы, участвующие в процессе оформления заявки и работы с тату-идеями: клиент, проверка возраста, заявка, тату-идея и референс-файл. Диаграмма наглядно показывает