

4.4 Диаграмма проектных классов

Диаграмма проектных классов предназначена для представления внутренней структуры разрабатываемой информационной системы в терминах программных классов и связей между ними. Каждый класс отображается в виде прямоугольника, разделённого на три секции:

- имя класса;
- атрибуты (поля/свойства), описывающие данные, которые хранятся в объекте;
- операции (методы), реализующие поведение объекта.

В отличие от диаграммы концептуальных классов, проектная диаграмма ориентирована на реализацию и учитывает архитектурные решения: наличие сервисных и контроллерных классов, идентификаторов, типов данных, а также операции, полученные при построении диаграмм взаимодействий и состояний.

На основании диаграммы последовательностей (рис. 4.2) и диаграммы состояний (рис. 4.3) были уточнены обязанности классов, после чего для каждого класса определён набор атрибутов, необходимых для хранения и обработки данных при учёте и проведении тренировок в фитнес-клубе SportFit. Сводный перечень атрибутов приведён в таблице 4.4.

Таблица 4.4 – Выделенные атрибуты проектных классов

Класс	Название атрибута	Назначение
Client	clientID	Уникальный идентификатор клиента в системе.
	name	Имя клиента.
	phone	Контактный номер телефона клиента.
	email	Адрес электронной почты для уведомлений.
ClientProfile	profileID	Уникальный идентификатор профиля.
	age	Возраст клиента.
	goal	Цель тренировок (похудение, набор массы, выносливость).
	level	Уровень подготовки (начальный, средний, продвинутый).

	restrictions	Медицинские ограничения и противопоказания.
Trainer	trainerID	Уникальный идентификатор тренера.
	name	ФИО тренера.
	specialization	Специализация (силовой тренинг, кардио, растяжка).
TrainingProgram	programID	Уникальный идентификатор программы.
	name	Наименование тренировочной программы.
	type	Тип программы (силовая, кардио, функциональная).
	duration	Длительность программы (в неделях).
Exercise	exerciseID	Уникальный идентификатор упражнения.
	name	Наименование упражнения.
	type	Тип упражнения.
	muscleGroup	Целевая группа мышц.
	load	Рекомендуемая нагрузка.
ExerciseCatalog	catalogID	Идентификатор каталога упражнений.
	totalExercises	Общее количество упражнений в каталоге.
TrainingPlan	planID	Уникальный идентификатор плана.
	startDate	Дата начала действия плана.
	endDate	Дата окончания действия плана.
	totalWorkouts	Общее количество тренировок в плане.

	approved	Флаг согласования плана клиентом.
Workout	workoutID	Уникальный идентификатор тренировки.
	date	Дата проведения тренировки.
	duration	Длительность тренировки.
	completed	Флаг выполнения тренировки.
ProgressReport	reportID	Уникальный идентификатор отчёта.
	period	Период, за который построен отчёт.
	weight	Показатель веса клиента.
	volume	Показатель объёмов.
	endurance	Показатель выносливости.
Membership	membershipID	Уникальный идентификатор абонеента.
	type	Тип абонеента.
	startDate	Дата начала действия абонеента.
	endDate	Дата окончания действия абонеента.
	active	Флаг активности абонеента.
Visit	visitID	Уникальный идентификатор посещения.
	date	Дата посещения.
	time	Время посещения.
	confirmed	Флаг подтверждения посещения.
ClubAdministrator	adminID	Уникальный идентификатор администратора.
	name	ФИО администратора.

	phone	Контактный телефон администратора.
SystemController	controllerID	Идентификатор сервисного компонента, отвечающего за расчёты и генерацию документов.

На основе перечисленных атрибутов и ранее определённых операций была построена диаграмма проектных классов, отражающая:

- классы предметной области (Client, ClientProfile, TrainingPlan, Workout, ProgressReport, Membership, Visit);
- сервисные классы и роли (Trainer, ExerciseCatalog, ClubAdministrator, SystemController);
- связи ассоциации, агрегации и зависимости между ними (заполнение профиля, подбор программы, формирование плана, учёт тренировок, расчёт прогресса, оформление абонеента, фиксация посещений).

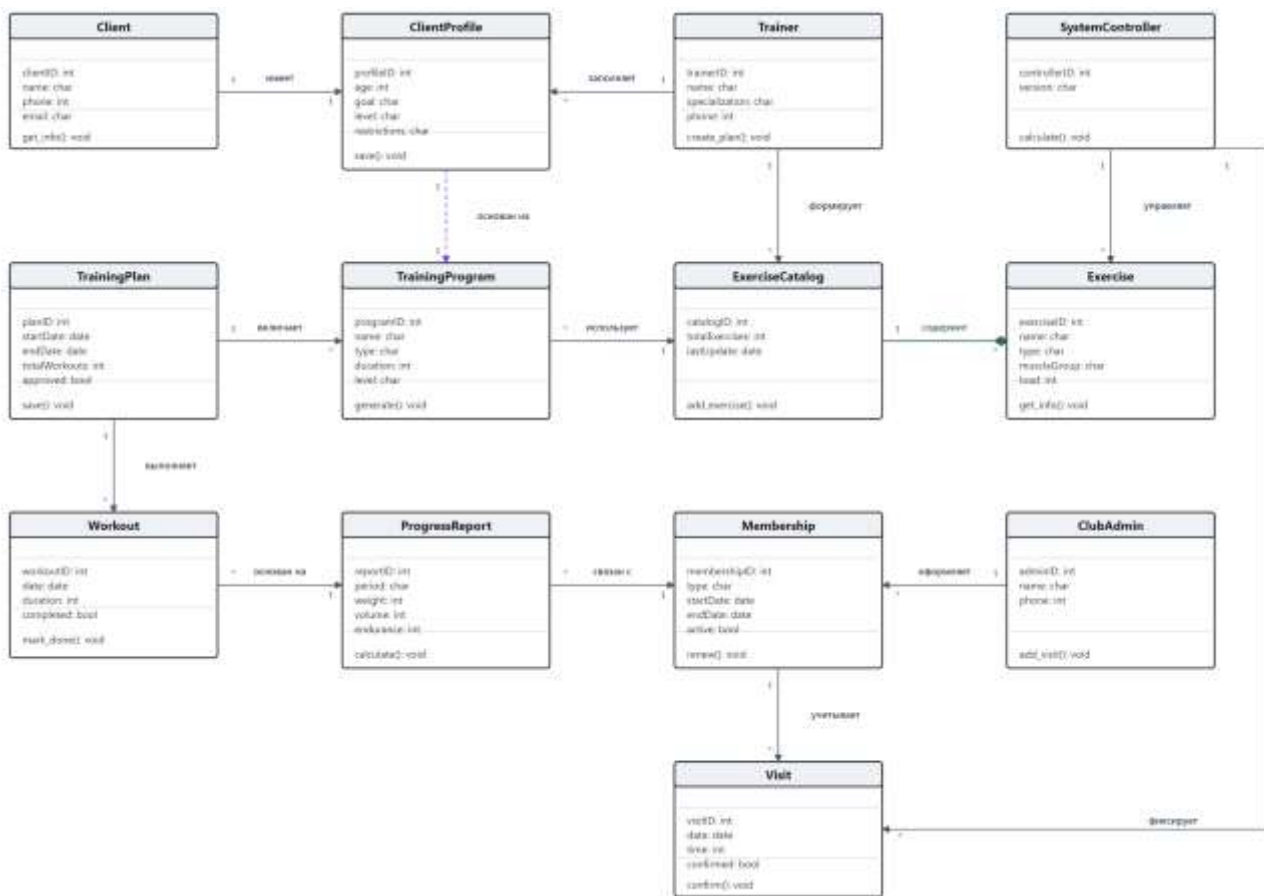


Рисунок 4.4 – Диаграмма проектных классов

Диаграмма проектных классов демонстрирует, как бизнес-объекты и сервисные компоненты совместно реализуют процесс учёта и проведения тренировок в фитнес-клубе SportFit и формируют основу для последующей реализации в виде классов языка программирования.