Princípios de POO utilizados:

Encapsulamento: Uso de private em todas as classes devido a cada cliente ter seus dados individuais e todas as classes possuírem get/set para possíveis visualizações e alterações nos dados.

Herança: Não aplicável

Polimorfismo: Temos os métodos toString para todas as classes, mas cada um corresponde a própria classe de sua maneira, embora o mesmo nome.

Classes e Objetos:

. Agendamento: Classe que representa um horário marcado, assim como se refere a pessoa a qual fez o agendamento e sobre qual pet é o agendamento no sistema. A classe também proporciona o método de cálculo de tempo e valor total, demonstrando como o sistema proporciona visualizações dos dados para maior entendimento do administrador.

. Persistência: Classe que representa a persistência dos dados do sistema, responsável pelas suas funcionalidades relacionados a dados. Serviços como autenticar e buscar por e-mail e cpf estão nessa classe, demonstrando encapsulamento sobre métodos que outras classes não podem ter acesso.

. Pet: Classe que representa os animais de cada cliente no sistema e seus atributos básicos, assim como gets e sets.

. Produto: Classe que representa os produtos do sistema e seus atributos básicos, assim como gets e sets.

. Serviço: Classe que representa os tipos de serviços disponíveis no sistema, diretamente ligado aos serviços disponíveis para agendamentos, assim como gets e sets.

. Usuário: Classe que representa o cliente/usuário do sistema com seus atributos básicos, assim como gets e sets.