Assesment Chipsoft

Om code-duplicatie te vermijden en gemeenschappelijke functionaliteiten te centraliseren, erven de klassen Patient en Physician over van de abstracte parentklasse Relation. Deze structuur zorgt voor herbruikbaarheid van code en voldoet aan het Single Responsibility Principle (SRP) doordat gedeelde eigenschappen en methoden binnen Relation worden beheerd.

Daarnaast worden PatientBuilder en PhysicianBuilder gebruikt om respectievelijk patiënten en artsen te creëren. Beide builders maken gebruik van de methode base.CreateRelation() uit de RelationService, wat ervoor zorgt dat gemeenschappelijke velden correct worden geïnitialiseerd. Dit ondersteunt het Open/Closed Principle (OCP), omdat uitbreidingen kunnen worden toegevoegd zonder bestaande code te wijzigen.

De RelationService beheert niet alleen de creatie van relaties (Patient en Physician), maar biedt ook functionaliteiten voor het weergeven en beheren van deze entiteiten, bijvoorbeeld bij het verwijderen, bij het plannen van afspraken of bij het ophalen van afspraken van een specifieke arts of patiënt. De AppointmentService behandelt alle logica rond afspraken en maakt gebruik van de klasse Appointment, waarin elk afspraak een Timeslot bevat. De Timeslot-klasse bevat een starttime en endtime en wordt hergebruikt in de Availability-klasse, die de beschikbaarheid van een arts weergeeft.

Bij het selecteren van een arts of patiënt wordt een uniek nummer gevraagd in plaats van een naam, om verwarring door duplicaten te voorkomen. Dit nummer is afgeleid van de Id, die een Guid is. Hierdoor blijft de identificatie consistent. Zowel AppointmentService als RelationService maken gebruik van interfaces, wat aansluit bij het Dependency Inversion Principle. Ook zit er zo goed als geen logica in de Program.cs file maar zit alles verwerkt in zijn desbetreffende service. Verder is er validatie verwerkt in de applicatie. Bijvoorbeeld bij het aanmaken bij een afspraak worden er allerlei checks uitgevoerd zoals zien of de ingegeven tijden vallen binnen een tijdslot, of de eindtijd na de starttijd is.