

Sprint 1 Testing

Georgie Parkin, Lily Bradshaw, James Aris, James Foulstone, Zack Njoroge

Test Description

1. Register a Surgery to the database with all columns filled in correctly.
2. Register a Doctor to the database with all columns filled in correctly.
3. Register a Patient to the database with all columns filled in correctly.
4. Register a Booking to the database with all columns filled in correctly.
5. Register a Booking with a DoctorID that is not in the database.
6. Register two surgeries with the same surgeryID but different names
7. Register a Doctor with all columns empty.
8. Change the surgeryID column in the doctor database for a specific doctor to a different one that exists.
9. Change the doctorID column in the Booking database for a specific booking to a different one.
10. Change a BookingTime in the Booking database for a specific booking to the same time it originally was.
11. Register a Surgery but give it a surgeryID that is a string rather than an integer.
12. Register a patient and insert the PatientPhoneNo, which should be one number short of the total length.
13. Register a Doctor but have the DoctorSpecialties blank.
14. Change a patient's details to be the same as another patient's except for NHSno.
15. Register a Booking instance but leave the BookingReason empty
16. Change the NHSno of a patient to a random string.
17. Change the BookingID of a booking instance to a BookingID held by another instance.
18. Remove a Surgery that some Doctors are registered at.
19. Remove a doctor instance that is booked for a booking instance.
20. Remove a patient who is in a booking instance.

What does it test?

1. This is to make sure that you can insert a surgery entity
2. This is to make sure that you can insert a doctor entity
3. This is to make sure that you can insert a patient entity
4. This is to make sure that you can insert a booking entity
5. To check what occurs when an entity is created with an incorrect piece of information.
6. This is to see what occurs when you attempt to register two entities with the same primary key.
7. This checks what happens when you attempt to register an entity into the database without any data.
8. Check what occurs when an attempt to change a piece of data occurs.

9. See what happens when a piece of data is changed.
10. See what occurs when a piece of data is changed to what it already is.
11. See what occurs when data is changed to a value it shouldn't be.
12. See what occurs when a piece of data, usually a specific length, is not that length.
13. See what occurs when a piece of entity is created without a piece of data
14. To see what occurs when two entities are identical except for the primary key.
15. To see what occurs when an entity is created without a piece of data.
16. To see what occurs when a piece of data is changed to a data type that it shouldn't be.
17. To see what occurs when the primary key of an entity is changed while being used as a foreign key in a different entity.
18. To see what occurs when an entity is deleted while its value is used as another entity's foreign key.
19. To see what occurs when an entity is deleted while its value is used as another entity's foreign key.
20. To see what occurs when an entity is deleted while its value is used as another entity's foreign key.

Was it successful

All the tests passed; the ones with incorrect information had errors, as expected. When the information provided was wrong, the system threw the expected errors. This indicates that the system is functioning as intended and can handle incorrect inputs gracefully. It is essential to ensure that the system can handle such scenarios to prevent unexpected behaviours and maintain the overall reliability of the system.