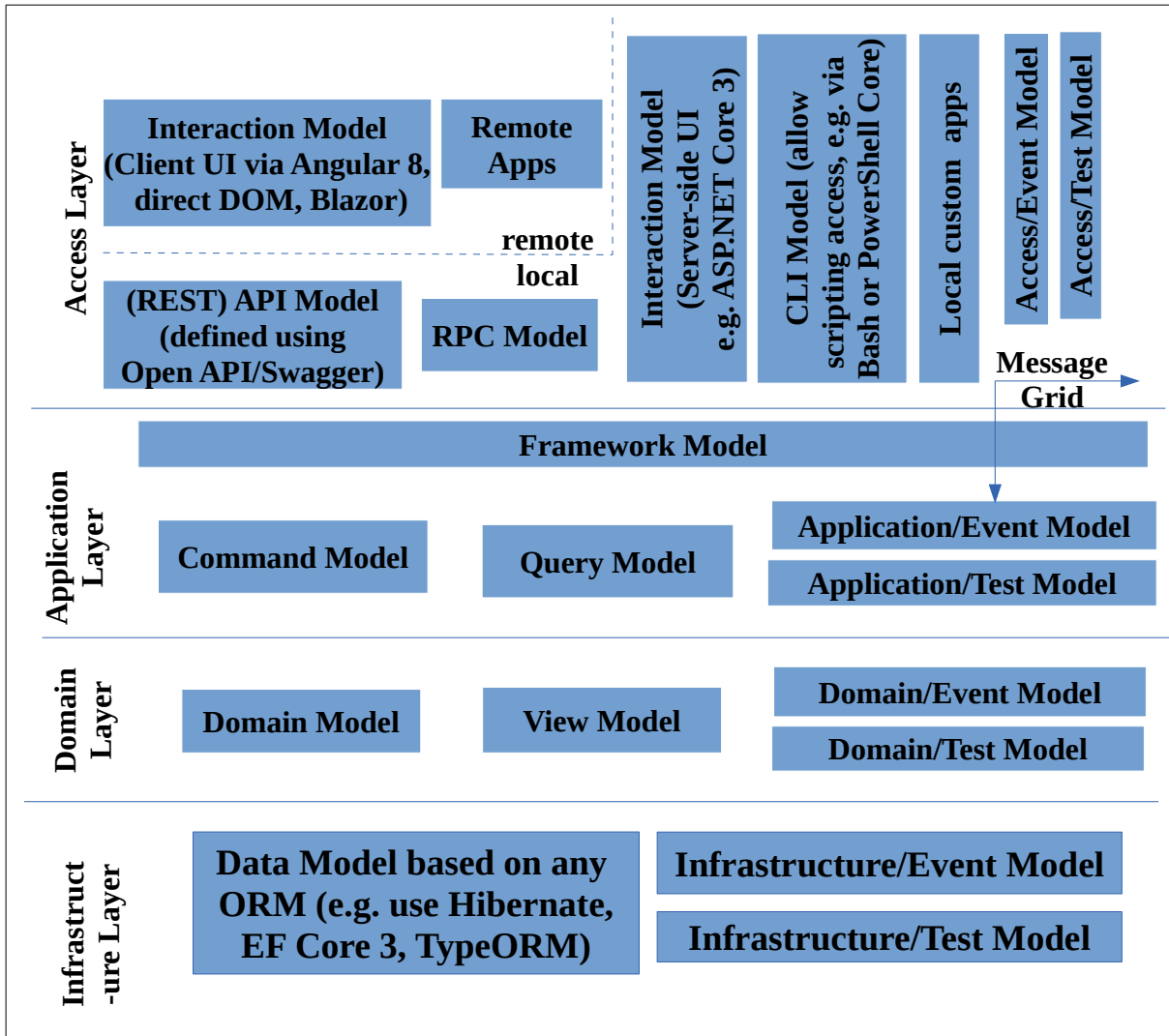


Web Portal Architectural Sketch (Technical Interview)

Q: Draw an outline of the architectural layout of a large web portal project.

A:



Notes: Domain Model should contain classes for entities, aggregate roots, domain services, factories, etc. It should also contain an interface for a bounded context and repositories (let's call them `IMyBoundedContext` and `IMyRepository`). Importantly, the domain model should not contain any database code or link in any way to database frameworks. The View Model is for simpler representations of resources, that are likely to be destined for display in the UI. They often are the result of a select query across multiple tables and is likely to have heavy use of text fields. The Data Model is where the actual database code resides and will be specific to a particular ORM. It will have concrete classes that implement `IMyBoundedContext` and `IMyRepository`. The data model links to the domain model, not vice versa. Heavily trafficked portal should consider supporting both REST API and gRPC (we see TensorFlow, which started with offering only a gRPC interface, now adding REST support; we see ASP.NET Core, which started with a REST API, now [also supporting gRPC](#)).