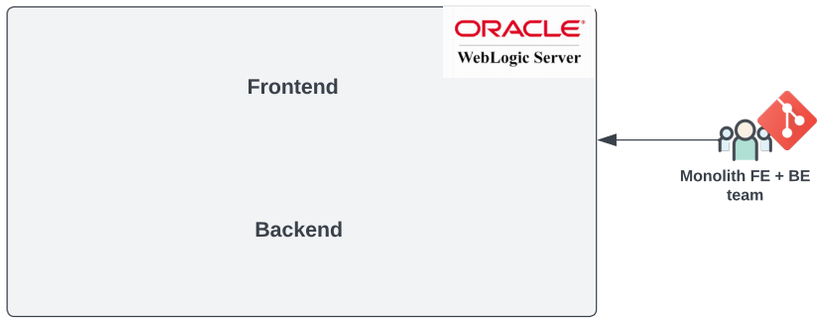
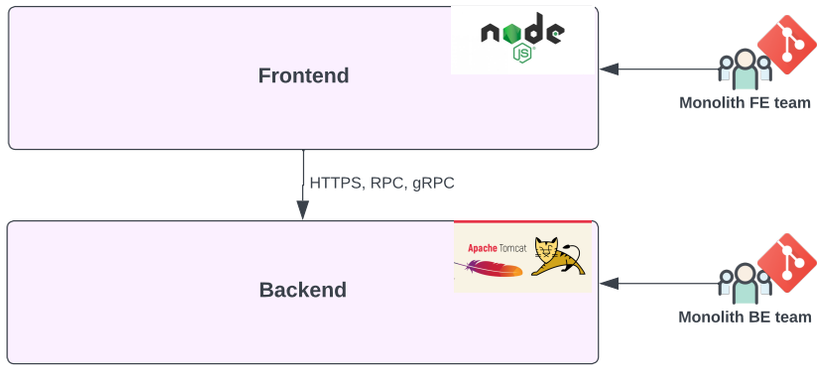
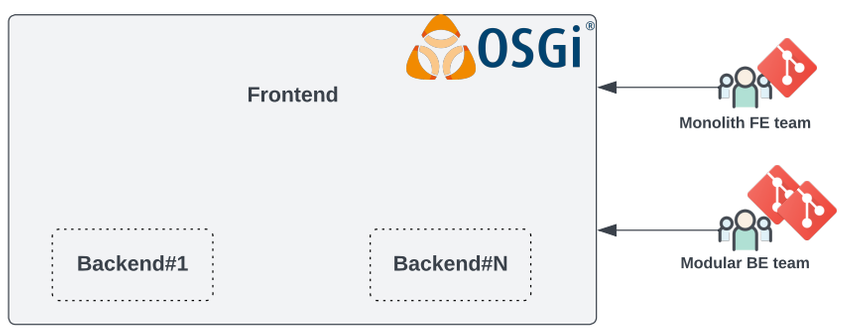
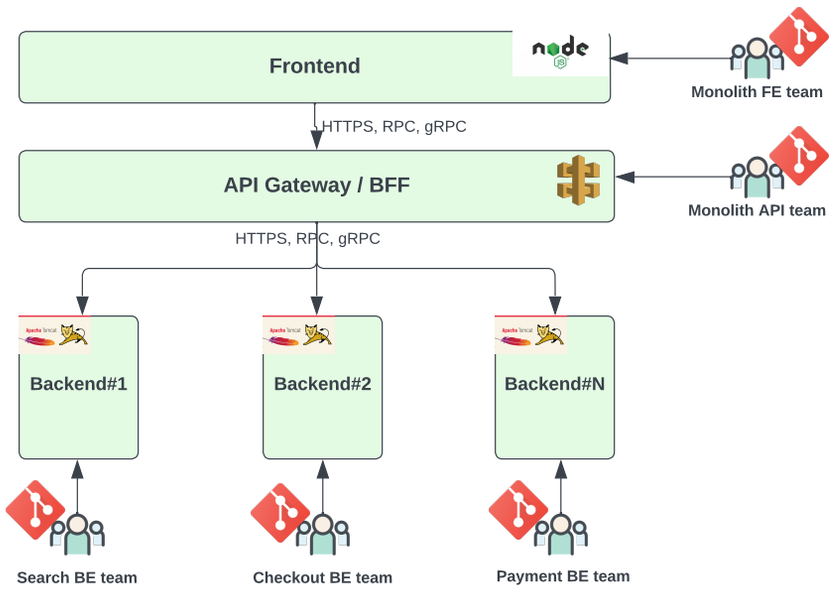
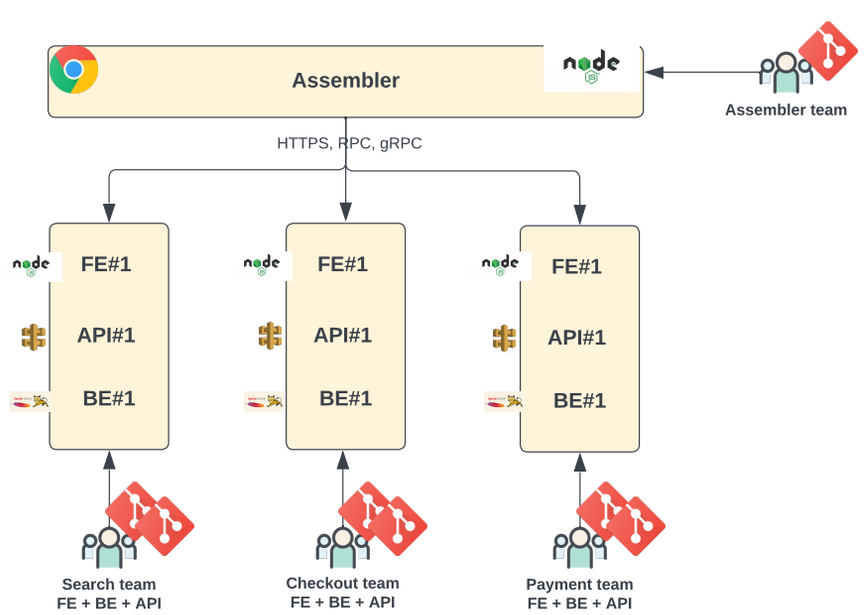
Type	Diagram	Details	Team structure	Release & Deployment	Testing scope
Monolith	 https://www.linkedin.com/in/vineet-kumar-765759b6/	All the code is developed and deployed collectively on a single server. Scaling require replicating the same server with entire code base making it expensive. One bad code can bring the entire system down e.g. memory leak in one of the component of the solution.	One big team comprises of all the tracks https://www.linkedin.com/in/vineet-kumar-765759b6/	Generally one CI/CD pipeline or process that deploys entire codebase during every release.	Generally most of the regression suite to be run since entire/most of the codebase was deployed in the same server.
Frontend/Backend Separation		Frontend pieces are separated from backend pieces and deployed into respective servers. Frontend communicates with Backend using HTTP(S), RPC or gRPC protocols. Frontend and backend severs can be scaled independently based on the demand. Issues in frontend and backend are isolated and hence better from resiliency point of view.	Frontend track is separated from backend track.	Different source code control for every track. Separate CI/CD pipelines make sure that respective code is properly build and deployed as per its track best practices. FE and BE tracks can have their independent release schedules.	Limited to the track whose code was just deployed.
Modular Monolith		This is still monolith as whole code is deployed in the same server. However code is structured, stored, build &/or packaged in a modular fashion e.g. OSGi container supporting hot deployment of modular Jars without impacting other jars. https://www.linkedin.com/in/vineet-kumar-765759b6/	Team is divided into separate logical tracks based on the structure of the project.	Deployment can be done by individual tracks as per their release schedule.	Even though deployment can be done by individual track, regression testing should still be in scope for other functionalities which are dependent on the deployed package since deployment is in the same server.
Microservices	 https://www.linkedin.com/in/vineet-kumar-765759b6/	Backend is divided into smaller, domain-driven microservices which are orchestrated and exposed through an API Gateway. These microservices are deployed into their respective servers & DBs which are best suited for their use cases. Frontend & API layer is still monolith. Micro services can be scaled easily based on the demand. Issue in one microservice is isolated from other microservices deployed in other servers. Bring in greater flexibility of tech stacks on backend.	Backend track is divided into separate teams (2 pizza teams) i.e. Search team, Checkout team etc. No division in FE and API team. https://www.linkedin.com/in/vineet-kumar-765759b6/	Individual backend team will have their own CI/CD pipelines and servers. They will release based on their own release schedule without impacting others. FE and API team will generally have one CI/CD pipeline respectively.	For backend code, only the functionality which was deployed requires to be tested, provided these services are following proper versioning. However for changes in FE or API, most of the FE and API functionalities are to be tested.
Microservices & Microfrontend		Taking microservices concept to both FE and API. Expensive to set-up and manage but can provide great value for clients requiring extreme agility. https://www.linkedin.com/in/vineet-kumar-765759b6/	A cross-functional team (FE+BE+API) is created for every functionality. Functionalities are broken down leveraging domain-driven design. An assembler framework layer is required to stitch the output of all teams together.	Every functionality can be released based on their schedule without impacting others resulting into higher agility. Advanced CI/CD pipelines make sure that code is deployed to respective servers. Assembler should be created as a framework and should change rarely else it can turn into monolith very soon. Assembler can be server-side or client-side based on project requirements.	Limited to the functionality just released.