Khi người dùng thực hiện yêu cầu tại UI, yêu cầu có một số cách thực hiện:

* Truyền yêu cầu đến keystone nếu auth token không thể bắt được:  
  thông tin được truyền bằng giao thức HTTP chứa thông tin auth trong header. Dựa vào thông tin trong header để Keystone trả lời lại UI bằng một token tạm thời.
* Truyền đến NOVA API nếu auth token chưa bị hết hạn  
  NOVA API cung cấp cho hệ thống khả năng quản lý nhiều API trên những domain con khác nhau.  
  +EC2-compatiple(tương thích)-bắt đầu được yêu cầu  
  +Compute API- tất cả sự đổi mới đều xảy ra ở đây
* Gửi yêu cầu API đến NOVA API  
  UI gửi yêu cầu POST đến NOVA API(được đánh đấu với những token nhận được)
* Xác nhận API token  
  NOVA API gửi yêu cầu thông qua HTTP để xác nhận API token đến keystone.  
    
  Người dùng tạo mật khẩu  
  họ dùng mật khẩu vừa tạo để thiết lập bộ đôi khóa  
  khóa public được đánh dấu và lưu trữ trong keystone  
  từ điểm này, người dùng sử dụng giấy chứng nhận tài khoản khách để đăng nhập  
  NOVA API thực hiện kiểm tra offline việc xác nhận token sử dụng CA&Cert mà nó có từ keystone.  
    
  keystone sẽ kiểm tra API token và trả lại phản hồi HTTP với thông tin token acceptance/rejection cho NOVA API .
* Chạy chương trình yêu cầu API  
  Xác định yêu cầu  
  +Lỗi thuật toán được kiểm tra trên cấp độ code  
  +mối quan hệ đám mây được xác định thông qua yêu cầu DB  
    
  Nếu yêu cầu không thể được thực thi thì ném một exception  
    
  Nếu yêu cầu có thể được thực thi  
  +Lưu trạng thái ban đầu vào database(NOVA database lưu trữ trạng thái gần nhất của tất cả object trong compute cluster)  
  NOVA API dựa vào yêu cầu và làm nó có hiệu lực bằng cách tìm kiếm dữ liệu từ NOVA DB. Nếu yêu cầu có giá trị, nó lưu trữ sự ra vào dữ liệu ban đầu của VM vào database.  
    
  +gửi tin nhắn với hành động tiếp theo đến MQ(Message Queue)  
  MQ là một cách hợp nhất giữa các thành phần NOVA  
  Ví dụ Message Process:   
  NOVA API  
  component A(tạo/update sự ra vào)🡪NOVA database  
  component A(gửi tin nhắn JSON với ID người nhận và ID đến, dợi đến bước tiếp theo)🡪RabbitMQ  
  NOVA Scheduler  
  component B(lựa chọn người nhận, lấy ID ra vào(entry ID) và thực thi nó trong DB)🡪RabbitMQ  
  component B(lấy ra vào(entry) bằng entry ID cho những chương trình xa hơn)🡪NOVA database  
    
  ở đây có 2 chế độ:  
  rpc.cast-không đợi kết quả(fire và forget)  
  rpc.call-đợi kết quả(khi có thứ gì đó trả về)
* Publish yêu cầu thật(provision)  
  NOVA API gửi rpc.call đến Scheduler. Nó publish một tin nhắn ngắn đến scheduler queue với thông tin VM.  
  yêu cầu được xác định, nhưng chưa có hành động nào được thực hiện
* Chọn yêu cầu thật(provision)  
  Schedeler sẽ chọn tin nhắn từ MQ
* Lập lịch provisioning  
  Scheduler tìm thông tin về toàn bộ cluster từ database, filter và lựa chọn compute node và update DB với ID của nó.
* Provision lập lịch  
  Scheduler publish tin nhắn đến compute queue (dựa vào host ID) để trigger VM provisionning
* Bắt đầu VM provisioning  
  NOVA compute có nhiệm vụ chính là khởi tạo và hoàn thiện những VM dựa trên hypervisor API  
  NOVA compute lấy tin nhắn từ MQ  
  NOVA compute tạo rpc.call đến NOVA conductor để lấy thông tin của VM từ DB  
    
  dịch vụ NOVA conductor là key để hoàn thiện no-db-compute
* Cấu hình network  
  NOVA compute gọi đến Quantum API để provision network ngay lập tức  
  Quantum cấu hình IP, gateway, DNS name, L2 connectivity,…(bằng cách gọi đến các node khác: static hoặc dynamic).
* Yêu cầu khối(volume)  
  với một volume đã được tạo rồi. NOVA compute liên hệ đến Cinder để lấy dữ liệu volume, hoặc cũng có thể gắn những volume này vào sau VM được xây dựng.  
  NOVA compute cài đặt iSCSI và chỉ dẫn Hypervisor để gắn iSCSI vol thành một thiết bị block mới.
* Yêu cầu VM Image từ Glance  
  NOVA compute yêu cầu VM Image dựa vào image ID
* Lấy Image URI từ Glance  
  nếu image với image ID nhận được dược tìm thấy thì trả về cho NOVA compute URI của image và sử dụng HTTP để truyền URI về cho NOVA compute.
* Download image từ Swift   
  NOVA compute download image tại Swift sử dụng URI nhận được từ Glance.
* Bắt đầu quá trình diễn tả (mô phỏng) VM dựa vào Hypervisor  
  NOVA compute tìm thông tin về VM từ DB, tạo ra một lệnh điều khiển đến hypervisor và delegate quá trình diễn tả VM đến hypervisor.  
  Nhờ vào KVM/ libvirtd, đây là một file cấu hình VM
* VM khởi động  
  NOVA compute gửi một tin nhắn đến NOVA conductor để update DB với trạng thái VM  
  việc giao tiếp iSCSI được bắt đầu với volume
* Người dùng cảm thấy hạnh phúc  
  Horizon poll NOVA API tình trạng của VM và trạng thái năng lượng mà được lấy từ database.