**P0: Opening**

Next, let me share technical solution part for Booking solution

In this part, we will cover 3 sub-topics, the 1st is Booking High Level Solution, the 2nd is Booking Application Architecture, the last one is Booking Integration Architecture.

Let’s move to the 1st one.

**P1: Booking High Level Solution**

For Booking application, we planned to develop it with popular web framework React, it can be built as SDK to integrate with existing mobile or web application, like EApp, also can be deployed as a standalone web application in Nginx web server.

* These blue flags indicate initiating booking process

1. Retail Adviser schedule booking to customer/operation staff from mobile or web app,
2. Booking Service persist data to database, then publish notification message to service bus.
3. Notification Service consume notification message from service bus and send SMS/Email to customer or operation staff.

Like below sample:

<You've scheduled an appointment with Eddie Wei, Greta Xu for 30 min meeting on 30 Jul 2024 at 02:45 PM (Asia/Singapore GMT +08:00) Number is CAIA-00003.

Something amiss? You can always reschedule or cancel your appointment.>

* These red flags indicate customers reschedule/accept/reject process.

1. Customer can accept or reject this booking. also, can reschedule the booking.
2. Booking Service persist data to database, then publish notification message to service bus.
3. Notification Service consume notification message from service bus and send SMS/Email to Retail Adviser or operation staff.

* This black flag with number 7, is system administration function.

1. Admin can manage user, config notification template, query booking history, view audit log, etc.

* This black flag with number 8, is Notification service integrate with SMTP service and SMS gateway, if need, we can extend to integrate with more communication mode.

**P2: Booking Application Architecture**

This is Booking application architecture diagram, it is a popular three-tier implementation, consists of a separated presentation tier, backend tier, and data tier.

1. Like what we said before, booking application is developed with React web framework. can be built as an SDK, and be integrated into existing mobile app and web app, like EApp.
2. Booking application also can be deployed separately in Nginx server in AKS to provide booking capabilities.
3. Admin portal web application to manage use, config notification template, review booking report, etc.
4. We use Azure APIM to manage and protect our exposed APIs.
5. Booking Service based on java and spring boot tech stack, is responsible to providing various booking functionalities.
6. Notification Service based on java and spring boot tech stack, is responsible to providing notification functionalities. it integrates with SMTP, SMS gateway. Communication mode can be extended if need.
7. Admin portal backend service. provide system administrate capabilities combine frontend application.
8. Introduce Azure Service Bus to decouple notification messages sending.
9. Persist data using Azure SQL Server.
10. Using Okta to authenticate booking users.

**P3: Booking Integration Architecture**

This diagram is integration architecture, it consists of the applications, backend services, networks, security, data, azure cloud services, and other dependent components that support the whole booking application environment, we use azure cloud to deploy booking application in this design.

1. This is existing application that integrate with Booking SDK
2. This is standalone Booking web application

For these 2-access approach, consider from security, because all users are from internet, so we need to access backend APIs per external API Gateway.

1. This is Admin portal application, because it is an internal admin tool, only access in intranet, we needn’t API Gateway.
2. We use AKS to deploy all booking application and services, booking namespace in AKS, hosts booking service, notification service and booking web application. All services in this namespace need to support user access from internet.
3. Admin Portal namespace in AKS, hosts admin portal service and admin portal web application. All services in this namespace need to support user access from intranet.
4. We use SQL Server database to store our business data, and use message bus to decouple notification message sending.
5. Other external dependencies, we need sync mobile calendar, and outlook calendar into booking application’s calendar, need to integrate SMTP service and SMS Gateway for user notification.
6. We use these azure services to monitor, observe and manage running application and services in azure cloud.