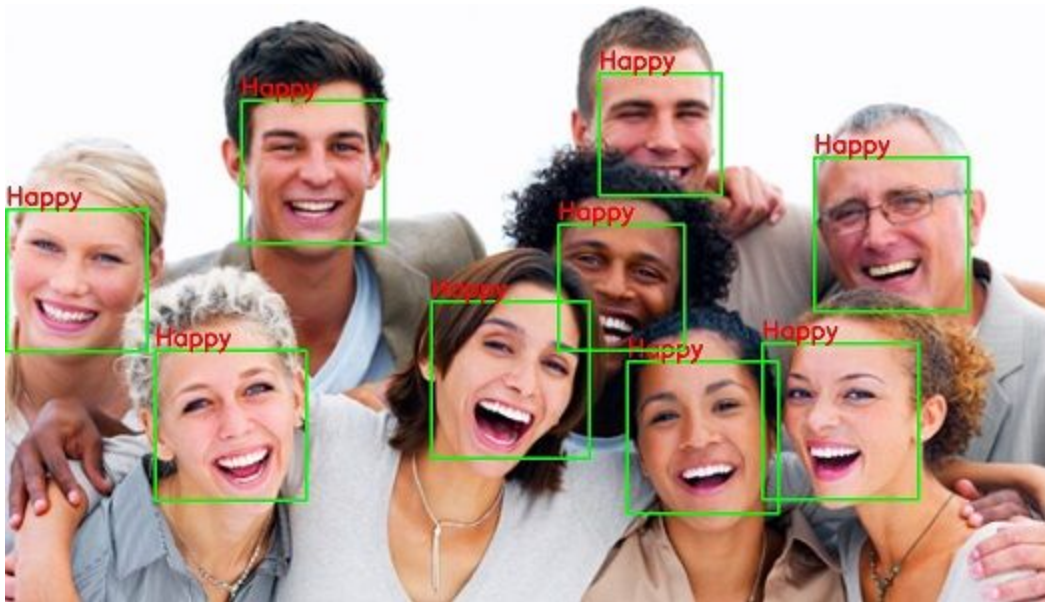


## Hotel Smart Service



### I. Aspectos generales

El objetivo de este proyecto es que el estudiante **individualmente** utilice todo el conocimiento a través del semestre para analizar y resolver un problema actuando como un arquitecto de software y desarrollador de software. El cliente del sistema a crear es una cadena hotelera que desea poder ofrecer a sus huéspedes una aplicación móvil que contribuya a mejorar la comunicación hotel-cliente durante su estancia, mejorando su experiencia y ofreciendo un servicio cinco estrellas. Además se desea que mejore la calidad de servicio para los huéspedes, que exista un nuevo canal de comunicación interactivo, mejorar la imagen de marca, que sirva como herramienta práctica de fidelización, y como plataforma de marketing para futuras acciones.

El cliente solicita que el sistema pueda resolver las siguientes necesidades:

- **Manejo de usuario:** Se desea poder registrar a los usuarios a la aplicación por medio de un formulario sencillo. Este formulario para registrar el usuario o para iniciar sesión no deben ser el primer punto de entrada a la aplicación. Solamente

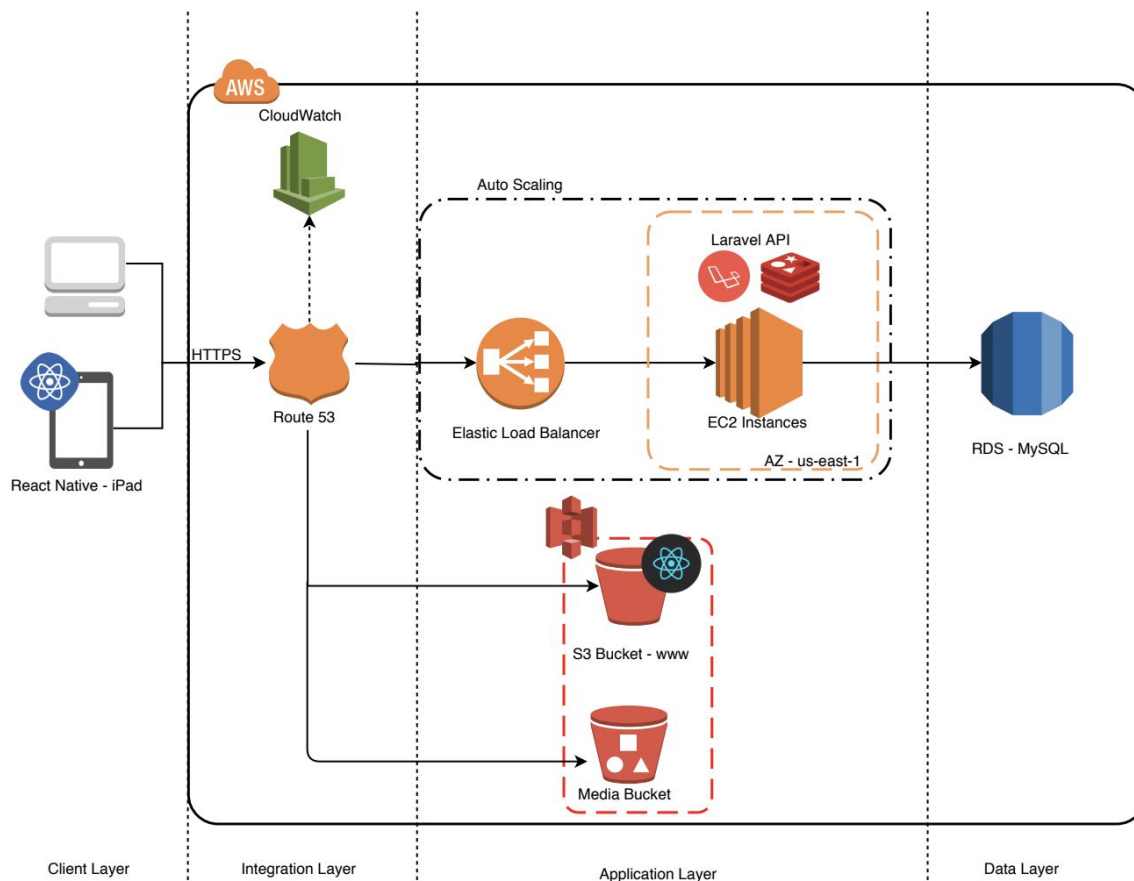
si se desea realizar alguna reserva o calificación se va a solicitar la autenticación.

- **Información del hotel:** Se desea poder captar a los clientes, mostrando información atractiva, imágenes, videos que permitan atraer al cliente a solicitar la reservación en el hotel.
- **Previsión local del tiempo:** Se desea poder mostrar el pronóstico del tiempo según las fechas solicitadas, para que el cliente pueda saber cuales son las mejores fechas para visitar la zona.
- **Detalles de los tipos de habitaciones:** Se desea mostrar la información de cada habitación, a través de imágenes y texto. Que muestre el número de camas y demás amenidades con las que cuenta cada tipo de habitación.
- **Servicios del hotel:** Se desea mostrar los servicios que ofrece el hotel como piscina, spa, restaurante, salas de conferencia etc.
- **Información de la zona:** Se desea mostrar la información de la zona en la que está el hotel, para que el cliente pueda saber cuáles actividades se pueden realizar en las cercanías como canopy, rafting etc.
- **Enlace a reserva online:** Este enlace debe abrir una página web donde el cliente puede realizar la reserva en línea. La implementación del frontend para el sistema de reservas no es parte del enunciado. Solamente el backend.
- **Enlaces a redes sociales:** Estos enlaces deben redireccionar al cliente a las diferentes páginas de twitter, facebook, instagram.
- **Multiidioma:** La aplicación debe soportar multilinguaje en español e inglés.
- **Experiencias de clientes:** Se desea tener un muro de publicaciones donde los clientes puedan subir las fotos y comentarios junto a una calificación que evidencie la experiencia durante la estadía. La calificación se obtiene por medio de un selfie y del servicio cognitivo en la nube que permita determinar la expresión facial y obtener una calificación de feliz, neutro, enojado o triste.
- **Contacto:** Debe mostrar la información de contacto del hotel.
- **Agenda de actividades:** Se desea mostrar el calendario de actividades del hotel, para que los clientes puedan saber a qué hora pueden acercarse a las zonas del hotel donde se van a realizar y además reportar el interés uniéndose. Por ejemplo, clases de aeróbicos, yoga.

- **Gestor de contenidos:** Se desea tener una herramienta que permita administrar todo el contenido que será mostrado día a día en la aplicación para todas las pantallas solicitadas.

## II. Arquitectura

El estudiante debe analizar la necesidad del cliente y diseñar una arquitectura de software que le permita soportar el sistema a crear. Un diagrama de arquitectura a nivel profesional luce de la siguiente manera.



Por lo tanto, se espera que utilicen este ejemplo como base para realizarlo. Usualmente se trabajan 3 capas, cliente, aplicación y datos. El ejemplo anterior agregó una capa de integración para evidenciar el manejo de red por medio de un firewall y router.

### III. Aspectos administrativos

1. Las tecnologías a escoger y toda la implementación queda a decisión del estudiante. Se espera que las pantallas de la aplicación sean atractivas para el usuario.
2. El diagrama de arquitectura se podrá revisar hasta el 30 de mayo.
3. La tarea es individual y se entrega el día sábado 15 de junio del 2019 en la cita de revisión **presencial** indicada por el profesor.
4. No se permitirá defensa por medio de video llamadas o algún otro medio. De no presentarse o llegar después de 15 minutos tarde pierde la totalidad del puntaje asignado a la prueba.
5. **Entrega:** Por medio del classroom. Subir el url del github y asegurarse de dar permiso al profesor para acceder al repositorio.
6. El repositorio debe contener el archivo Readme.md, dónde se documenta como instalar todo lo necesario para ejecutar el programa y alguna otra indicación que se considere importante. Recuerde que esto es documentación pertinente a un desarrollador de software.
7. Cualquier duda, omisión o contradicción en la especificación se debe aclarar con el profesor.
8. Cualquier clase de copia de código será sancionada de acuerdo con el reglamento vigente y se llevará hasta la consecuencia de amonestación con carta al expediente.
9. Si en la defensa no demuestra su autoría con el dominio propio de esa calidad se le asignará una nota de cero en la totalidad de la prueba.
10. El código debe estar suficientemente documentado de tal forma que usted se pueda orientar en él fácilmente durante la defensa.
11. La documentación a entregar es una reflexión de máximo dos hojas que comente sobre cómo selecciona, aplica y evalúa con criterio técnicas, recursos o herramientas modernas de ingeniería para la solución del problema propuesto.

### IV. Evaluación

Tanto el lado cliente como backend deben estar completamente funcionales cumpliendo correctamente todas las necesidades del cliente.

- Cliente 12%
- Backend 12%
- Documentación 1%