

BeatIt!

Informe de Verificación de Integración

Versión 1.2.2



Historia de revisiones

Fecha	Versión	Descripción	Autor
08/09/2014	1.2.1	Completado inicial	Emiliano Vázquez
12/09/2014	1.2.2	Revisión de SQA	Pablo Olivera

Contenido

- ["Hola mundo!" Prueba de Interfaz](#)
 - [Requerimientos Funcionales](#)
 - [Aspecto](#)
 - [Rapidez](#)
 - [Requerimientos No funcionales](#)
 - [Interacción en la iteración](#)
- ["Usain Bolt" Prueba de manejo API GPS](#)
 - [Requerimientos Funcionales](#)
 - [Seguimiento del dispositivo](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- ["Bouncing game!" Prueba de comprensión de la API del acelerómetro](#)
 - [Requerimientos Funcionales](#)
 - [Movimiento del dispositivo](#)
 - [Requerimientos No Funcionales](#)
- ["Selfie" Prueba de comprensión y de compatibilidad para reconocimiento facial](#)
 - [Requerimientos Funcionales](#)
 - [Reconocimiento facial](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- ["Es una botella de Coca-Cola?" Prueba de comprensión y de compatibilidad de API de reconocimiento de objetos](#)
 - [Requerimientos Funcionales](#)
 - [Reconocimiento de objeto](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- ["Ritmos y notas" Prueba de comprensión y de compatibilidad de APIs de reconocimiento de sonido](#)
 - [Requerimientos Funcionales](#)
 - [Análisis de sonido](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- ["Exprésate!" Prueba de comprensión para manejo de funcionalidad de envío de sms](#)
 - [Requerimientos Funcionales](#)
 - [Envío de mensaje de texto](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- ["WebServer" Prueba de distintas opciones de servidor web](#)
 - [Requerimientos Funcionales](#)
 - [Plataforma Microsoft Azure](#)
 - [Tiempo de respuesta](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
 - [Integración con repositorio GitHub](#)
- ["Database Server" Prueba de persistencia de información](#)
 - [Requerimientos Funcionales](#)
 - [Plataforma Microsoft Azure](#)
 - [Tiempo de respuesta](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- ["Servicio Rest" Prueba de implementación servicio rest](#)
 - [Requerimientos Funcionales](#)
 - [Tiempo de respuesta](#)
 - [Requerimientos No Funcionales](#)
 - [Interacción en la Integración](#)
- [Métricas de pruebas cubiertas](#)

1. "Hola mundo!" Prueba de Interfaz

Se instaló el ambiente de desarrollo de Appcelerator Titanium, en el cual generamos una pequeña prueba del estilo "hola mundo", percatándonos que no se trabajaba con código nativo de la plataforma, jugando como una contra a la hora de querer trabajar a un nivel más detallado de implementación.

[Por mayor detalle documento IMPRG6 punto 1.1]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
UI móvil	Prueba de Interfaz	Martín Alayón Martín Berguer
...	...	

1.1. Requerimientos Funcionales

- Las interfaces gráficas de cada plataforma deben tener los estilos de cada una.
- El tiempo que transcurre entre que un jugador realiza una acción y tiene la posibilidad de hacer otra acción, no debe superar los 3 segundos.

1.1.1. Aspecto

Entrada	Salida esperada	Salida obtenida
Ninguna	Interfaz presentando un mensaje con aspecto que siga la línea de la plataforma	Colores, layout e interacciones respetan la plataforma. El mensaje se presentó correctamente.
...

1.1.2. Rapidez

Entrada	Salida esperada	Salida obtenida
Ninguna	Transiciones rápidas	En general se apreció una rápida respuesta al input generado por el usuario.
...

1.2. Requerimientos No funcionales

N/A

1.3. Interacción en la iteración

N/A

2. "Usain Bolt" Prueba de manejo API GPS

Ante las pruebas realizadas se obtuvo comprensión del servicio GPS brindado por las dos plataformas, además si validó los desafíos planteados y relacionados.

[Por mayor detalle documento IMPRG6 punto 1.2]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
Lógica móvil	Desafío Usain Bolt	Luciana Martinez Cristian Bauza Joaquin Velazquez
...	...	

2.1. Requerimientos Funcionales

-Desafío del primer prototipo

2.1.1. Seguimiento del dispositivo

Entrada	Salida esperada	Salida obtenida
Captura del GPS	Coordenadas acordes a la región y velocidad en tiempo real	Coordenadas reportadas, válidas, velocidad medible con precisión exacta
...

Errores encontrados	Sugerencias de corrección
La señal del GPS es débil bajo techo Nivel de gravedad: Marginal	Los desafíos elegidos con uso del GPS serán especialmente pensados para realizarse al aire libre, puesto que el error es debido a una limitante de hardware.
...	...

2.2. Requerimientos No Funcionales

N/A

2.3. Interacción en la Integración

N/A

3. "Bouncing game!" Prueba de comprensión de la API del acelerómetro

La prueba fue satisfactoria ya que se lograron los objetivos deseados, en primer lugar, obtener los datos del acelerómetro y en segundo interpretarlos correctamente para lograr que cuando (por ejemplo) balanceamos el celular a la derecha una esfera en la pantalla fuera a la derecha.

[Por mayor detalle documento IMPRG6 punto 1.3]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
--------------------------------	---------------	----------------------

Lógica móvil	Desafío Bouncing game!	Luciana Martinez Cristian Bauza Joaquin Velazquez
...	...	

3.1. Requerimientos Funcionales

-Desafío seleccionado para el producto final

3.1.1. Movimiento del dispositivo

Entrada	Salida esperada	Salida obtenida
Captura del acelerómetro	Mediciones de aceleración del dispositivo móvil	Las mediciones fueron obtenidas para los distintos ejes espaciales permitiendo conocer el movimiento del dispositivo.
...

3.2. Requerimientos No Funcionales

N/A

3.3. Interacción en la Integración

N/A

4. "Selfie" Prueba de comprensión y de compatibilidad para reconocimiento facial

El resultado fue positivo ya que se encontró un webservices para el reconocimiento facial. A su vez se obtuvo la ventaja de que un webservice soporta la dos plataformas.

También se realizó para Android una prueba para el reconocimiento facial, se concluyó que es posible realizarlo localmente ya que se consiguió una librería para dicho proposito.

[Por mayor detalle documento IMPRG6 punto 1.4]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
Lógica móvil	Desafíos Selfie Groupie & Sacate una selfie	Luciana Martinez Cristian Bauza Joaquin Velazquez Martín Steglich
...	...	

4.1. Requerimientos Funcionales

-Desafío seleccionado para el producto final

4.1.1. Reconocimiento facial

Entrada	Salida esperada	Salida obtenida
---------	-----------------	-----------------

Imagen capturada por cámara	Identificar cuantos rostros se encuentran en una imagen	Es posible reconocer cuantos individuos y donde se encuentran en la imagen.
...	...	

4.2. Requerimientos No Funcionales

N/A

4.3. Interacción en la Integración

N/A

5. "Es una botella de Coca-Cola?" Prueba de comprensión y de compatibilidad de API de reconocimiento de objetos

Se realizó con éxito la captura de imágenes desde el dispositivo, en base a documentación disponible en internet.

En cuanto a la investigación de APIs locales o webservices para el reconocimiento de objetos, encontramos limitaciones de presupuesto, y para las opciones libres requería un nivel de implementación más elevado por lo que implicaba un riesgo de curva de aprendizaje más prolongado para validar este desafío.

[Por mayor detalle documento IMPRG6 punto 1.5]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
Lógica móvil	Desafío Es una botella de Coca-Cola?	Luciana Martinez Cristian Bauza Joaquin Velazquez
...	...	

5.1. Requerimientos Funcionales

-Desafío seleccionado para el producto final

5.1.1. Reconocimiento de objeto

Entrada	Salida esperada	Salida obtenida
Imagen capturada por cámara	Etiquetas identificatorias de objeto en imagen	Inaccesible la información dado el costo
...	...	

5.2. Requerimientos No Funcionales

N/A

5.3. Interacción en la Integración

N/A

6. "Ritmos y notas" Prueba de comprensión y de compatibilidad de APIs de reconocimiento de sonido

La investigación de APIs locales o webservices para el procesamiento de sonido no tuvo éxito dado que no se encontraron opciones gratuitas que realizará el procesamiento de sonido tal como lo necesitábamos.

Se descarta la implementación de nuestro propio componente de procesamiento de sonido debido a la carencia teórica del tema.

[Por mayor detalle documento IMPRG6 punto 1.6]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
Lógica móvil	Desafíos Alcanzame una nota! & Seguime el ritmo	Luciana Martinez Cristian Bauza Joaquin Velazquez
...	...	

6.1. Requerimientos Funcionales

-Desafío seleccionado para el producto final

6.1.1. Analisis de sonido

Entrada	Salida esperada	Salida obtenida
Input grabado por el micrófono	Frecuencias de sonido grabadas	Información suficiente pero incomprendible por el equipo de desarrollo
...	...	

6.2. Requerimientos No Funcionales

N/A

6.3. Interacción en la Integración

N/A

7. "Exprésate!" Prueba de comprensión para manejo de funcionalidad de envío de sms

Resultado fue positivo ya que se encontró una librería que permite el acceso a los contactos del usuario (y obtener el número telefónico) y otra librería con la cual es posible enviar mensajes de texto.

Se encontró una forma de realizar el envío pasando por la aplicación de mensajes del teléfono, pudiéndose editar el mensaje y el contacto elegido, y también sin pasar por la misma.

[Por mayor detalle documento IMPRG6 punto 1.7]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
Lógica móvil	Desafíos Exprésate!	Martín Steglich

...	...	
-----	-----	--

7.1. Requerimientos Funcionales

-Desafío seleccionado para el producto final

7.1.1. Envío de mensaje de texto

Entrada	Salida esperada	Salida obtenida
Texto del mensaje y agenda de contactos	Envío de mensaje a contacto seleccionado	Se constato el envío correcto del sms
...	...	

7.2. Requerimientos No Funcionales

N/A

7.3. Interacción en la Integración

N/A

8. "WebServer" Prueba de distintas opciones de servidor web

La instalación fue muy sencilla teniendo un "Hola Mundo" rápidamente, tanto en una instalación de CentOS alojado en una máquina virtual en Microsoft Azure como directo en un Web Site.

[Por mayor detalle documento IMPRG6 punto 2.1]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
Lógica web	Procesamiento en la nube	Gonzalo Javiel
...	...	

8.1. Requerimientos Funcionales

-Procesamiento en Microsoft Azure

8.1.1. Plataforma Microsoft Azure

Entrada	Salida esperada	Salida obtenida
Web request a servidor NodeJs	Web response	Web response válida y puesta en marcha sencilla
Web request a máquina virtual con CentOS	Web response	Web response válida

8.1.2. Tiempo de respuesta

Entrada	Salida esperada	Salida obtenida
Web request	Json response	El formato Json es más liviano y por tanto más rápido
...	...	

8.2. Requerimientos No Funcionales

N/A

8.3. Interacción en la Integración

8.3.1. Integración con repositorio GitHub

Entrada	Salida esperada	Salida obtenida
Push de código al repositorio	Actualización automática del servidor web	El repositorio GitHub y el servicio WebSite quedaron vinculados perfectamente
...	...	

9. "Database Server" Prueba de persistencia de información

Prueba satisfactoria, se optó por probar una base de datos NoSQL luego de investigar la bondades de la misma en cuanto a los tiempos de respuesta, lo "natural" de la conexión con un servidor NodeJS y como es la definición del modelo. [Por mayor detalle documento IMPRG6 punto 2.2]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
DataStorage	Persistencia en la nube	Gonzalo Javiel
...	...	

9.1. Requerimientos Funcionales

-Persistencia en Microsoft Azure

9.1.1. Plataforma Microsoft Azure

Entrada	Salida esperada	Salida obtenida
Databse query	Datos persistidos	Datos correctamente persistidos en base de datos NoSQL
...	...	

9.1.2. Tiempo de respuesta

Entrada	Salida esperada	Salida obtenida
---------	-----------------	-----------------

Database query	Respuesta a las consultas rápida	La conexión al servidor NodeJS fue exitosamente rápida en comparación a otras tecnologías
...	...	

9.2. Requerimientos No Funcionales

N/A

9.3. Interacción en la Integración

N/A

10. "Servicio Rest" Prueba de implementación servicio rest

Como resultado se optó por utilizar una API RESTfull (ante las opciones WCF, ASP.Net, Web API, SOAP), debido a la extensa documentación en lo que respecta a la interacción con dispositivos móviles, la velocidad de respuesta por ser una implementación liviana en comparación a las otras tecnologías y por interoperabilidad con las otras herramientas elegidas.

[Por mayor detalle documento IMPRG6 punto 2.3]

Verificador:

Componentes o subsistemas que lo integran

Componente o subsistema	Nombre	Implementador
BeatIt! API	Integración de componentes	Gonzalo Javiel
...	...	

10.1. Requerimientos Funcionales

-Comunicación ligera entre componentes

10.1.1. Tiempo de respuesta

Entrada	Salida esperada	Salida obtenida
Web request	Respuesta rápida	Propio del estilo arquitectónico Rest la respuesta fue liviana ignorando sintaxis de protocolos, consideramos esta opción óptima para comunicación con dispositivos móviles.
...	...	

10.2. Requerimientos No Funcionales

N/A

10.3. Interacción en la Integración

N/A

11. Métricas de pruebas cubiertas

Aún quedan 4 desafíos por probar, al día de hoy las pruebas de integración hechas cubren suficiente para construir el prototipo 1 y 2 sin alto riesgo de encontrar problemas técnicos.

Por último a fines de la semana 4 comienzan las pruebas de configuración del sistema de integración continua.