



Departamento de Informática
Universidad Técnica Federico Santa María



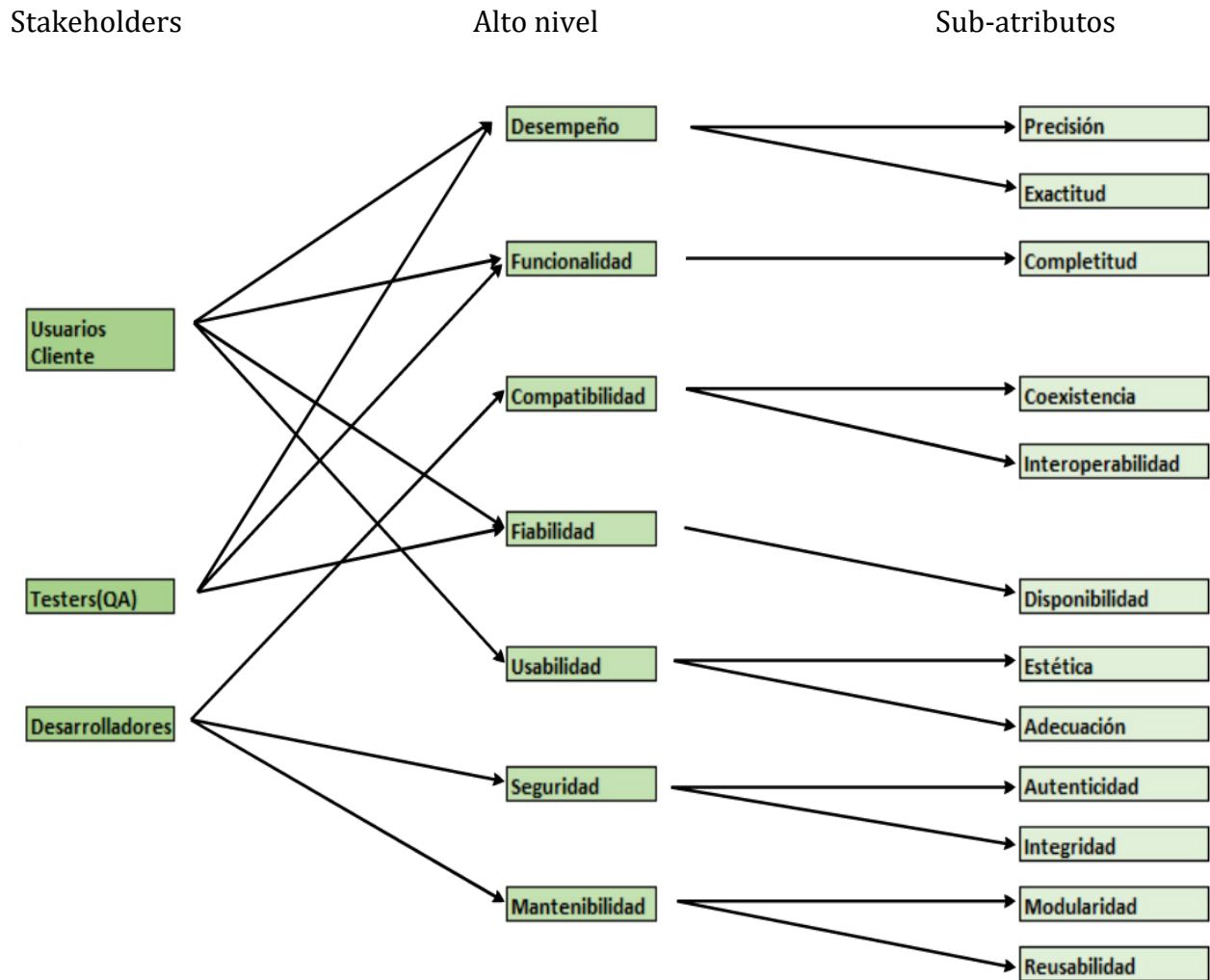
Entregable III

Proyecto: Classer

Integrantes:

Nombres y Apellidos	Email	ROL USM
Francisco García	francisco.garciam.13@sansano.usm.cl	201373540-1
Rodrigo Valenzuela	rodrigo.valenzuem.13@sansano.usm.cl	201373602-5
Simón Contreras M	simon.contreras.13@sansano.usm.cl	201323503-4

Modelo de Calidad



basado en ISO 25010.

Stakeholders:

Usuarios/Cliente: Los futuros usuarios y el cliente, el cual forma parte de éstos, indicó que una de las cualidades de la plataforma fuera el entrenamiento en conjunto del clasificador de texto, por lo cual, forman parte de los stakeholders del proyecto ya que sin ellos la plataforma no podría progresar, en el caso de que fuera puesta en producción.

Testers (QA): Se consideran stakeholders del proyecto ya que su objetivo es intentar mejorar la calidad del software, para así el proyecto tenga más probabilidad de éxito. Es por esto que están interesados en aspectos como la funcionalidad, desempeño y fiabilidad del sistema.

Desarrolladores (DEUX): Nosotros como grupo somos otro stakeholder del proyecto, ya que como el proyecto forma parte de las evaluaciones del ramo, debemos cumplir con los entregables estipulados como parte de la evaluación del curso. Además de poder cumplir con las expectativas del cliente/usuario de la plataforma ya que somos los creadores y encargados de la plataforma en desarrollo.

Atributos:

Desempeño: como el enfoque del proyecto se basa en clasificar texto de forma automática, e ir mejorando este aspecto progresivamente, el desempeño del método implementado es primordial para poder satisfacer de forma adecuada el objetivo principal propuesto como grupo de desarrollo, en donde la precisión y exactitud condicionan directamente a este atributo.

- **Precisión y Exactitud:** características que se complementan entre sí, las cuales describen de forma compacta el comportamiento que se espera lograr con el clasificador de texto, durante una etapa madura del desarrollo de la plataforma, estas dos características resumen lo que se ha definido como desempeño para nuestro proyecto.

Funcionalidad: capacidad del software para proporcionar funcionalidades tanto explícitas como implícitas, cuando el software es utilizado bajo condiciones esperadas. Se identifica este atributo como de calidad, debido al proceso de desarrollo seguido mediante entregables donde se espera ir añadiendo más funcionalidades de forma progresiva.

- **Completitud:** cualquier funcionalidad o conjunto que no cumpla o que realice tareas parciales en la plataforma se considera no completa, lo cual impacta de forma negativa el atributo de funcionalidad, por ello, el poder marcar una funcionalidad o conjunto como completadas se considera satisfactorio y aporta de forma positiva a la plataforma en desarrollo.

Compatibilidad: el desarrollar una plataforma web que pueda estar ejecutándose en un computador sin comprometer tanto el rendimiento en general de éste como del navegador en el que está siendo utilizada, se considera un atributo no menor para nosotros como desarrolladores, ya que se busca que la plataforma pueda ser utilizada en la mayor cantidad de dispositivos y sistemas operativos posibles.

- **Coexistencia e Interoperabilidad:** la plataforma debe poder funcionar en presencia de procesos independientes en el computador donde se esté ejecutando ya sea de forma local como mediante un navegador de internet, y coexistir sin afectar el rendimiento de las demás aplicaciones. Además, el uso de servicios externo, los cuales son utilizados por los módulos de extracción y clasificación requieren que el atributo de interoperabilidad se cumpla, ya que estos servicios son esenciales e irremplazables en la plataforma.

Fiabilidad: Para poder generar un proceso de entrenamiento estable del clasificador y del crecimiento en la colaboración de la página, ésta debe poder cumplir con las funcionalidades en un entorno estable ya sea a nivel local como en un host en internet.

- **Disponibilidad:** para cumplir con el atributo de fiabilidad, se establece que la disponibilidad de la plataforma en línea durante el desarrollo, es primordial tanto para poder comprobar el funcionamiento de la plataforma, su continuo perfeccionamiento, y que cliente vaya observando los avances progresivos que se realizan sobre el proyecto.

Usabilidad: el cliente indicó como un requisito deseable, que la interfaz para la plataforma fuera fácil de utilizar, y con una estética amigable e intuitiva, por lo cual este atributo se considera como uno de los con mayor prioridad respecto a la calidad del software en desarrollo.

- **Estética y Adecuación:** como se indicó anteriormente la plataforma debe poseer la capacidad de agradar y satisfacer la interacción con el usuario, además la plataforma debe ser capaz de mostrar al usuario si ésta es adecuada para satisfacer las necesidades que este posee.

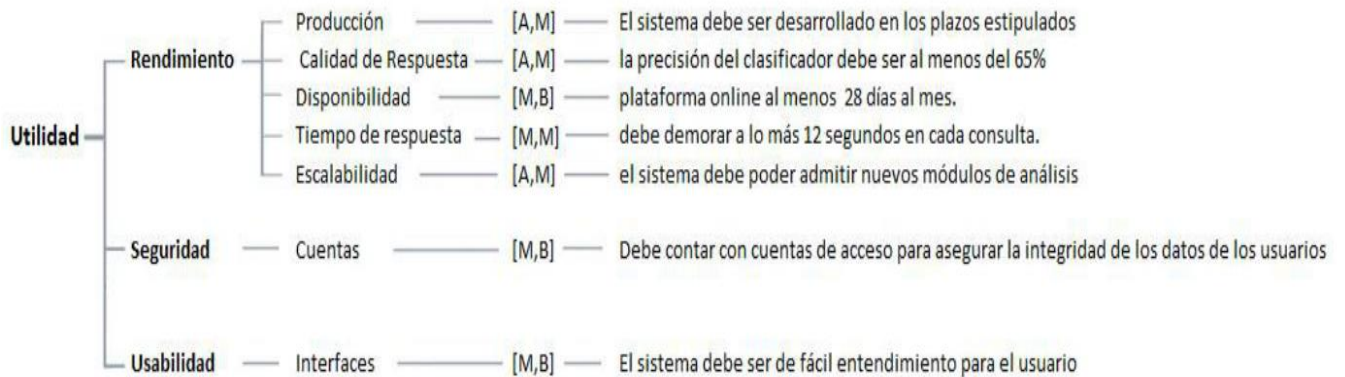
Seguridad: como requisito se solicitó controlar el acceso al sitio mediante cuentas de acceso y que la información recopilada sea almacena en algún tipo de repositorio, base de datos, etc. Por lo tanto, el acceso a la plataforma (Autenticidad) y la integridad de datos (Integridad) es relevante para el funcionamiento de la plataforma.

Mantenibilidad: cualquier observación por parte de los testers o por parte del cliente que requiera modificar el código de la plataforma y con ello sus funcionalidades, debe repercutir lo menos posible en la estructura general de la implementación, ya que cualquier componente debe poder ser reparable o actualizable sin comprometer a los

demás de forma radical y con esto no dejar inutilizable la plataforma.

- **Modularidad y Reusabilidad:** tanto la modificación de algún componente existente como la integración de uno nuevo, debe ser de forma modular, para no intervenir de forma global la plataforma y con ello mantener las funcionalidades aisladas, sin comprometer todo. En caso de tener que reformular casos de usos o módulos en específico se busca que el esqueleto base pueda ser reutilizados en futuras actualizaciones de funcionalidades o de mayor escala.

Árbol de utilidad (actualización)



Con respecto al primer árbol de utilidad, se tiene que los principales cambios corresponden a:

- **Eliminación de Vulnerabilidad en la rama de Seguridad:** el tiempo de desarrollo restante para el proyecto y la poca prioridad inicial que poseía esta rama, la hace prescindible en esta etapa del proyecto por lo cual es eliminada del árbol.
- **Eliminación de la rama de Confiabilidad:** el considerar el sub-atributo de latencia como parte de confiabilidad, se considera impreciso, por lo cual se opta por eliminar la rama, para no producir confusión, además la incertidumbre de no saber si la plataforma será llevada a producción hace irrelevante considerar este atributo en el árbol.

Con respecto a las ramas eliminadas estas no han afectado de forma directa al proyecto, ya que la naturaleza de éstas y su verdadera aplicación se vería realmente en etapas finales del desarrollo.

- **Cambio en el escenario de Calidad de Respuesta:** luego de recibir el feedback del cliente se establece que los perfiles cognitivos ya no son preponderantes en la plataforma a desarrollar, en donde ahora el enfoque es poder aumentar la precisión y exactitud del clasificador a lo menos al 65%, la cual se pretende aumentar en primera instancia, mediante entrenamientos supervisados. Con respecto al desarrollo del proyecto, estaba contemplado el realizar entrenamientos supervisados, por lo cual no ha implicado ni una mejora ni retraso en el desarrollo.
- **Cambio escenario Disponibilidad:** como la plataforma en desarrollo está siendo levantada en un host online, se pretende realizar mantenciones las cuales no dejen la plataforma menos de 28 días al mes disponible para su uso. El anterior escenario era excesivamente irrealista, ya que las mantenciones a la

plataforma en estado de desarrollo son inevitables, por cual el considerar que “siempre esté disponible”, no aplica a la naturaleza del proyecto. El tener de antemano las fechas de cada entregable, proporciona una ventana de tiempo adecuada para poder realizar las mantenciones necesarias antes de poder tener la versión actualizada de la plataforma ya sea para testing de los QA o evaluación de los ayudantes, además mantener la página disponible para su uso común, por lo cual no ha afectado de manera drástica al proyecto.

- **Cambio escenario tiempo de respuesta:** se eleva el tiempo de respuesta por consulta a la plataforma de 10 segundos a 12 segundos, debido a la cantidad de llamadas a servicios externos que deben realizar el clasificador y el extractor de entidades, con una conexión a internet estable, se cumple este escenario en el 90% de los casos. No aplica un efecto negativo sobre el desarrollo del proyecto, ni tampoco sobre la experiencia del usuario en la plataforma.
- **Cambio escenario escalabilidad:** el primer escenario propuesto no se ajustaba al verdadero significado de Escalabilidad, por lo cual, se cambia por: “el sistema debe poder admitir nuevos módulos de análisis”, en donde si se aplica que a medida que se agreguen nuevos módulos como el analizador de imágenes, si se espera que la plataforma escale sin problemas. En la etapa del desarrollo en que se encuentra la plataforma no se han encontrado inconvenientes para agregar nuevos módulos.

Pruebas de Software

Descripción y justificación Pruebas realizadas:

Nombre caso de prueba:	Verificación de clasificación de texto vacío.
Descripción:	El académico utiliza el sistema para clasificar un texto, el cual se encuentra vacío.
Justificación:	Es importante saber si el sistema no fallará en momentos no esperados, como puede ser el descuido de un usuario al simplemente apretar un botón sin llenar el campo de texto antes.
Estado caso de prueba:	Exitoso

Nombre caso de prueba:	Verificación de clasificación correcta de un ejercicio de 2 preguntas.
Descripción:	El académico utiliza el sistema para clasificar un texto, el cual contiene 2 preguntas.
Justificación:	Es importante saber si el sistema no clasificará mal al momento de presentarse ante un ejercicio con dos preguntas y no arrojará una clasificación incorrecta.
Estado caso de prueba:	Exitoso

Nombre caso de prueba:	Verificación de clasificación correcta de un ejercicio de 1 pregunta
Descripción:	El académico utiliza el sistema para clasificar un texto con 1 sola pregunta.
Justificación:	Es atingente conocer si el sistema no responderá mal al momento de presentarse ante un ejercicio de esta índole y no arrojará una clasificación incorrecta.
Estado caso de prueba:	Fallido

Nombre caso de prueba:	Verificación de visualización de material
Descripción:	El académico ve el material ya parametrizado.
Justificación:	La página debe ser capaz de mostrar todo tipo de contenido.
Estado caso de prueba:	Fallido

Nombre caso de prueba:	Verificación de entidades principales de un texto
Descripción:	El académico desea obtener las entidades de un texto.
Justificación:	La página debe ser capaz de poder obtener los tópicos presentes en cada tipo de texto que se le presente
Estado caso de prueba:	En construcción

Planes de mejora:

Prueba realizada:	Verificación de clasificación correcta de un ejercicio de 1 pregunta
Causa del error:	Poca cantidad de ejemplos presentados
Plan de mejora:	Mejorar clasificador por medio de ejemplos

Prueba realizada:	Verificación de visualización de material
Causa del error:	No implementación del scrap de videos, ni html correspondiente.
Plan de mejora:	Implementar scrap de videos y html correspondiente.

Resultados obtenidos:

ID Caso de Prueba	Mejora Realizada	Resultado Prueba Inicial	Resultado Prueba post-mejora
CP1_ClasText	Se han añadido nuevos ejemplos al clasificador.	Exitoso	Exitoso
CP2_ClasText	Se han añadido nuevos ejemplos al clasificador.	Exitoso	Exitoso
CP3_ClasText	Se han añadido nuevos ejemplos al clasificador.	Fallido	Exitoso
CP5_VerMatPar	Se ha implementado el scraping de videos	Fallido	Exitoso
CP6_ObteEntiText	Ninguna	En construcción	En construcción
CEF9_Dispatch	Ninguna	Exitoso	Exitoso

Evaluación QA:**¿Cómo ha sido la participación del consultor QA en el proyecto?**

La participación de los consultores QA ha ido de menos a más, han sabido responder de mejor forma ante nuestros requerimientos, inquietudes y exigencias a medida que ha transcurrido el semestre. Si bien en un principio la coordinación y la comunicación fue un poco lenta debido a la carga académica que todos tenemos durante este periodo del semestre, se ha logrado sobrellevar y se pudieron desarrollar las pruebas sobre la plataforma de forma eficiente y oportuna.

¿Han sido un aporte para el proyecto las pruebas realizadas?

Si, las pruebas realizadas han servido para darnos cuenta de errores que hemos cometido que han derivado en defectos de software, provocando desviación en el resultado esperado. Gracias a esto hemos podido solucionar algunos de estos defectos y mejorar el software en desarrollo.

Como grupo, ¿Cómo calificaría el trabajo de su consultor QA hasta el momento? Fundamente su respuesta. Utilizar la siguiente escala->1: Muy deficiente, 2: Deficiente, 3: Regular, 4: Bueno y 5: Muy bueno.

Consideramos que el trabajo de los QA ha sido bueno, dado que han terminado mostrando responsabilidad y han propiciado una mejora en el proyecto. Lo cual beneficia a las dos partes involucradas.

Formulación de pruebas de software de requisitos no funcionales

Nombre caso de prueba	Verificación calidad de respuesta
Descripción	El sistema debe clasificar con una precisión de al menos 65%
Pasos y condiciones de ejecución	<ul style="list-style-type: none"> • ENTRADA: Variados textos de prueba • CONDICIONES DE EJECUCIÓN: El académico ingresa a la pestaña “Clasificar Texto”, luego ingresa el texto y presiona el botón “Clasificar”
Resultado esperado	La página debe clasificar al menos 65% de los textos de prueba correctamente

Nombre caso de prueba	Verificación tiempo de respuesta
Descripción	El sistema no debe demorar más de 12 segundos en clasificar un texto
Pasos y condiciones de ejecución	<ul style="list-style-type: none"> • ENTRADA: Variados textos de prueba • CONDICIONES DE EJECUCIÓN: El académico ingresa a la pestaña “Clasificar Texto”, luego ingresa el texto y presiona el botón “Clasificar”
Resultado esperado	La página debería clasificar todos los textos de prueba en un tiempo igual o inferior a 12 segundos

Nombre caso de prueba	Verificación de disponibilidad en un host
Descripción	El sistema debe estar en un host, para poder entrenar la máquina de aprendizaje.
Pasos y condiciones de ejecución	<ul style="list-style-type: none"> • ENTRADA: “http://simoncm.pythonanywhere.com/” • CONDICIONES DE EJECUCIÓN: El usuario con conexión a internet ingresa en la URL del navegador la dirección de la página web.
Resultado esperado	La página debe cargar en el navegador

Nombre caso de prueba	Verificación autenticación requerida
Descripción	Los usuarios del sistema deben ser autenticados para acceder a las distintas páginas.
Pasos y condiciones de ejecución	<ul style="list-style-type: none"> • ENTRADA: URLs del sistema • CONDICIONES DE EJECUCIÓN: El usuario ingresa alguna URL sin previa autenticación
Resultado esperado	La página debería pedir que el usuario se autentique para poder acceder

Nombre caso de prueba	Verificación de entidades principales de una imagen
Descripción	El sistema debe poder obtener las entidades de una imagen.
Pasos y condiciones de ejecución	<ul style="list-style-type: none"> • ENTRADA: Imagen • CONDICIONES DE EJECUCIÓN: El académico ingresa a la pestaña “Encontrar Entidades de Imagen”, luego ingresa una imagen y presiona el botón “Obtener Entidades”
Resultado esperado	La página debería mostrar las entidades más relevantes relacionadas a una imagen.

Listado de Pruebas/Requisitos

ID	Nombre del Requerimiento	Roles asociados	Descripción
RF1	Clasificación de texto	1) Académico	<p><u>Académico:</u> El académico puede utilizar el sistema para clasificar un texto.</p> <p>Una vez el académico solicite clasificar un texto, el sistema deberá mostrar si corresponde a una definición o un ejercicio.</p>
RF2	Obtención de Entidades	1) Académico	<p><u>Académico:</u> El académico puede utilizar el sistema para obtener las entidades de un texto.</p> <p>Una vez el académico solicite obtener entidades, el sistema deberá mostrar una lista con las palabras más relevantes de texto.</p>
RF3	Visualizar el material parametrizado	1) Académico	<p><u>Académico:</u> El académico puede ver el material parametrizado que ha sido recopilado, filtrando por tipo de material.</p> <p>Una vez el Académico ingrese a la pestaña material, el sistema mostrara el material parametrizado que ha sido</p>

			recopilado, siendo filtrado por el tipo seleccionado.
RF4	Disponibilidad en un host	No aplica	El sistema debe estar disponible en un host.

- RT1: Si texto fue ingresado. Clasificar.
- RT2: Si texto fue ingresado. Obtener entidades.
- RT3: Si ningún texto fue ingresado. Clasificar.
- RT4: Página actual: Clasificar Texto.
- RT5: Página actual: Encontrar Entidades.
- RT6: Página actual: Material.
- RT7: Filtro seleccionado: Videos.
- RT8: Usuario Logueado.
- RT9: URL Válida.

Caso de Prueba #	RF #	RT#	Caso de Prueba	Pasos del caso	Datos utilizados	Resultados obtenidos
CP1_ClasText	RF1	RT3 RT4 RT8	Verificación de clasificación de texto vacío	1) Ingresar a la pestaña Clasificar Texto 2) Ingresar Texto 3) Click en Clasificar	Texto= ""	No muestra clasificación (Exitoso)
CP2_ClasText	RF1	RT1 RT4 RT8	Verificación de clasificación correcta de un ejercicio de 2 preguntas	1) Ingresar a la pestaña Clasificar Texto 2) Ingresar Texto 3) Click en Clasificar	Texto= "Si el radio de una circunferencia es igual a 4 ¿Cuál es su diámetro? ¿cuál es su perímetro?"	Clasificación: Ejercicio (Exitoso)
CP3_ClasText	RF1	RT1 RT4 RT8	Verificación de clasificación correcta de un ejercicio de 1 pregunta	1) Ingresar a la pestaña Clasificar Texto 2) Ingresar Texto 3) Click en	Texto= "Si el radio de una circunferencia es igual a 4 ¿Cuál es su diámetro?"	Clasificación: Definición (Fallido)

				Clasificar		
CP4_VerMatPar	RF3	RT6 RT7 RT8	Verificación de visualización de material	1) Ingresar a la pestaña Material 2) Seleccionar Videos	Ninguno	Sin resultados (Fallido)
CP5_ObteEntiText	RF2	RT2 RT5 RT8	Verificación de entidades principales de un texto	1) Ingresar a la pestaña Encontrar Entidades 2) Ingresar Texto 3) Click en Obtener Entidades	Texto= "La paradoja de Fermi es la aparente contradicción que hay entre las estimaciones que afirman que hay una alta probabilidad de que existan otras <u>civilizaciones inteligentes</u> en el <u>universo observable</u> , y la ausencia de evidencia de dichas civilizaciones. Surgió en 1950 en medio de una conversación informal del físico <u>Enrico Fermi</u> con otros físicos del laboratorio pero ha tenido importantes implicaciones en los proyectos de búsquedas de señales de civilizaciones extraterrestres (SETI)."	<ul style="list-style-type: none"> • Contradicción • Civilizaciones • <u>Enrico Fermi</u> • Ausencia • Evidencia • Paradoja • Estimaciones • Probabilidad • Universo • Señales • Medio • Laboratorio • Físico • Conversación • Implicaciones • Proyectos • Búsqueda • SETI
CEF9_Dispon	RF4	RT9	Verificación de disponibilidad en un host	1) Ingresar URL	URL="http://simoncm.pythonanywhere.com/"	El sistema debe estar disponible al acceder al dominio.