



Diseño de Alto Nivel

23/03/2022

Versión 1.3

Historial de revisiones

Autores:

Darien Miguel Sánchez Arévalo

Angel Alejandro Salinas García

Fernando De Luna Guardiola

Fecha	Versión	Descripción	Autor
12/02/2022	1.0	Elaboración Inicial del Diseño de Alto Nivel	Fernando De Luna Guardiola
13/02/2022	1.1	Correcciones de Información	Darien Miguel Sánchez Arévalo
14/02/2022	1.2	Corrección ortográfica y de redacción	Angel Alejandro Salinas García
17/03/2022	1.3	Correcciones	Fernando De Luna Guardiola

Índice

1. Especificación de requerimiento de software	4
1.1 Introducción	4
1.2 Propósito	4
1.3 Alcance	4
1.4 Definiciones, Acrónimos y Abreviaturas	4
2. Descripción General	7
2.1 Perspectiva del Producto	7
2.2 Funciones del Producto	7
2.3 Objetivo del Diseño	7
3. Arquitectura	8
3.1 Introducción	8
Figura 1 Diagrama General	9
Tabla 1 Direccionamiento de red	10
3.2 Datos	11
Figura 2 Mapa E-R	11
3.3 Sistema Operativo	12
3.4 Comunicación	13
3.4.1 Usuario - Página Web	13
3.4.2 Página Web - Base de Datos	13
3.4.3 Máquina Virtual – Cliente	13
3.5 Módulos	14
3.5.1 Interfaz Gráfica del Usuario (Vistas)	14
3.5.2 Controlador	14
3.5.3 Acceso a Datos (Modelos, DAO)	14
3.6 Puertos de Enlace y DNS	15
4. Desarrollo	16
5. Apéndices	17
5.1 Conformidad con Estándares	17
5.2 Corrección de Errores	17
5.3 Seguridad	17

1. Especificación de requerimiento de software

1.1 Introducción

En este diseño de alto nivel (HLD) se provee una visión general del presente documento, el cual incluye: un propósito general, su alcance y una lista de definiciones, abreviaturas y referencias que serán de ayuda al analizar este documento, así mismo, se estará viendo toda la información para el desarrollo de una página web y una base de datos. Todo está desarrollado de acuerdo con las especificaciones que el cliente provee.

1.2 Propósito

El diseño de alto nivel describe en conceptos generales la arquitectura del proyecto que se va a construir. Abarcando desde: el desarrollo, el software que se usará, así como otros factores requeridos para brindar una descripción adecuada, entendible y completa.

1.3 Alcance

En este documento se describe la arquitectura del software a desarrollar, con el propósito de dar solución a la problemática que fue presentada por el cliente.

1.4 Definiciones, Acrónimos y Abreviaturas

- **HLD (High Level Design):** Explica la arquitectura que se utilizará para desarrollar un producto de software. El diagrama de arquitectura proporciona una descripción general de un sistema completo, identificando los componentes principales que se desarrollarán para el producto y sus interfaces.
- **Diagrama E-R:** Tipo de diagrama de flujo que ilustra como las entidades (como personas, objetos o conceptos) se relacionan entre sí dentro de un sistema.

- **HTML (HyperText Markup Language):** Lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web.
- **JavaScript:** Es un lenguaje de programación. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente del lado del cliente, implementado como parte de un navegador web, permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.
- **jQuery:** Librería de JavaScript cuyo objetivo es facilitar la programación scripting al ofrecer una serie de funciones y métodos con los cuales los usuarios pueden producir páginas web o sitios web más rápido y fácil.
- **PHP:** Lenguaje de código abierto especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. El código PHP suele ser procesado en un servidor web por un intérprete PHP implementado como un módulo.
- **HTTP (Hypertext Transfer Protocol):** Es un protocolo de la capa de aplicación para la transmisión de documentos hipermedia, como HTML. Fue diseñado para la comunicación entre los navegadores y servidores web, aunque puede ser utilizado para otros propósitos también.
- **HTTPS (HyperText Transfer Protocol Secure):** Protocolo de comunicación de Internet que protege la integridad y la confidencialidad de los datos de los usuarios entre sus ordenadores y el sitio web.
- **DNS:** Sistema de nomenclatura jerárquico descentralizado para dispositivos conectados a redes IP, como Internet o una red privada. Este sistema asocia información variada con nombres de dominio asignados a cada uno de los participantes.
- **TCP:** Protocolo más utilizado en Internet. Está orientado a la conexión, es decir, los datos pueden enviarse de forma bidireccional una vez establecida la conexión.
- **RDP:** Protocolo desarrollado por Microsoft que permite la comunicación en la ejecución de una aplicación entre una terminal (mostrando la información procesada que recibe del servidor) y un servidor Windows.

- **Bootstrap:** Es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML Y CSS.
- **Microsoft Azure:** Es un servicio de computación en la nube creado por Microsoft para construir, probar, desplegar y administrar aplicaciones y servicios mediante el uso de sus centros de datos.
- **DAO (Data Access Object):** Es un componente de software que suministra una interfaz común entre la aplicación y uno, o más, dispositivos de almacenamiento de datos, tales como una Base de datos o un archivo.
- **MySQL:** Es un sistema de gestión de bases de datos relacional y es considerada como la base de datos de código abierto más popular del mundo.
- **MCV (Modelo Vista Controlador):** es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización.
- **Backend:** Su función es acceder a la información que se solicita, a través de la aplicación, para luego combinarla y devolverla al usuario final.
- **ECMAScript:** Es el estándar que, a partir del año 2015 a la actualidad, se encarga de regir como debe ser interpretado y funcionar el lenguaje JavaScript.
- **SPL (Standard PHP Library):** Provee un conjunto de soluciones a situaciones comunes relacionadas con el recorrido de listas de archivos en un directorio, arreglos y filas devueltas por una consulta a una base de datos, o cualquier clase de colección.

2. Descripción General

2.1 Perspectiva del Producto

El producto que estamos manejando, es un sistema que fue desarrollado para el cliente, el cual tiene la intención de entretener a los usuarios, en donde podrán compartir sus puntuaciones y posicionarse en la tabla de mejores puntuaciones.

Se pretende utilizar este sistema para, a través de actualizaciones, agregar más elementos, así como funcionalidades, facilitar la interacción con un portal en línea, así como la administración de la información de los usuarios en una base de datos.

2.2 Funciones del Producto

Garden Rush es un videojuego, el cual cuenta con tres mapas distintos, cada uno de ellos cuenta con diferentes niveles y un jefe final. Los usuarios podrán probarlo sin ninguna restricción, inclusive teniendo la posibilidad de comenzar una partida con dos jugadores.

Otros usuarios podrán ver la cantidad de puntos que los demás jugadores vayan obteniendo, y estos mismos se estarán posicionando en una tabla, la cual contará con las mejores puntuaciones de los jugadores.

2.3 Objetivo del Diseño

El diseño que se verá a continuación está hecho para que haya una idea de cómo queremos implementar y desarrollar nuestro sistema, así como ver la manera en la que operaría una vez que todo esté terminado y funcionando. Todo de acuerdo con las indicaciones y especificaciones que nos solicitó el cliente.

3. Arquitectura

3.1 Introducción

Utilizaremos el servicio de Azure para crear una máquina virtual donde se desarrollará y hospedará la aplicación web. Contará con un sistema operativo de Windows, al mismo tiempo, éste tendrá una conexión con una base de datos externa que se hospedará también en los servicios de Azure. Se ha establecido, sin embargo, que se requiere de una división lógica entre 3 partes específicas del producto:

Como se ha mencionado anteriormente, utilizaremos una base de datos para almacenar la información que se genere del sistema, para esto se utilizará una base de datos en MySQL.

Por otra parte, el sistema contará con un backend que se encargará de comunicarse con la base de datos y así realizar el intercambio entre la información almacenada.

Finalmente, la interfaz gráfica de usuario, que se encargará de la captura de la información mediante los eventos que el mismo usuario activará.

Figura 1 Diagrama General

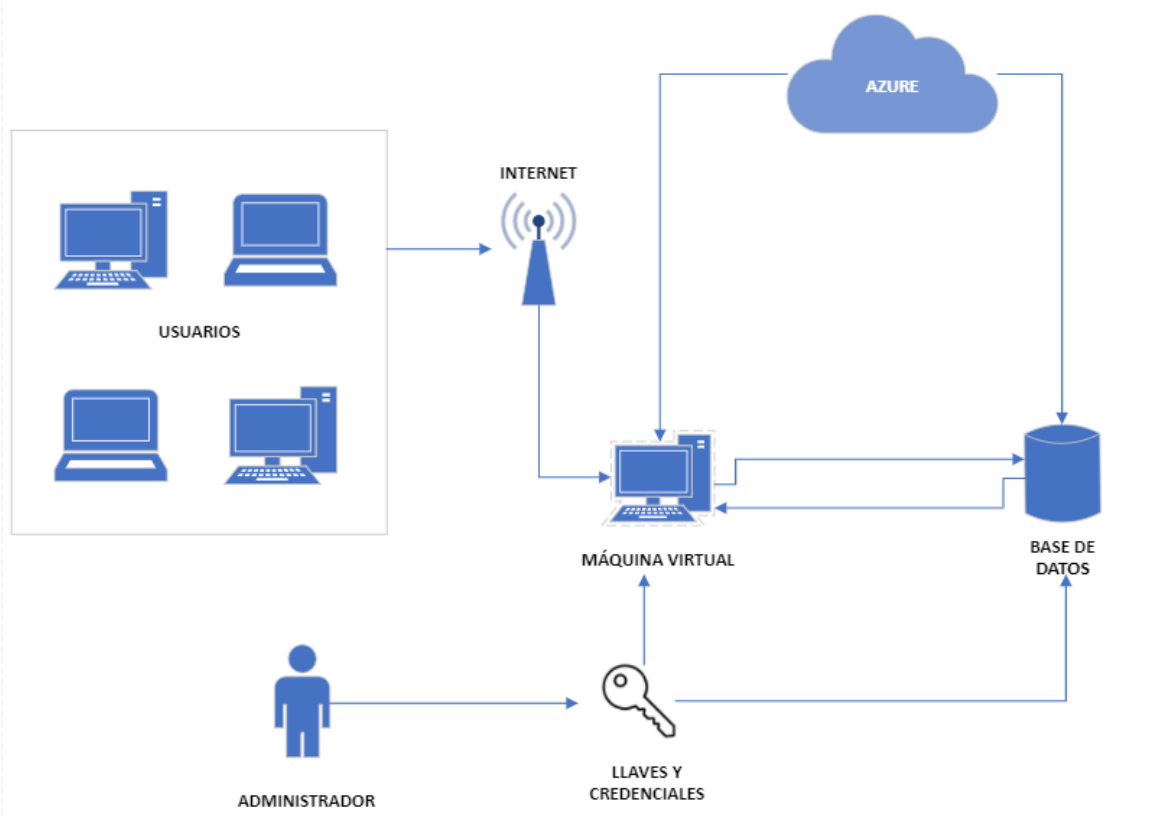


Tabla 1 Direccionamiento de red

20.127.114.154/11

Red de Clase: A 255.0.0.0 Máscara
Número de subredes: 4 255.224.0.0 Máscara personalizada

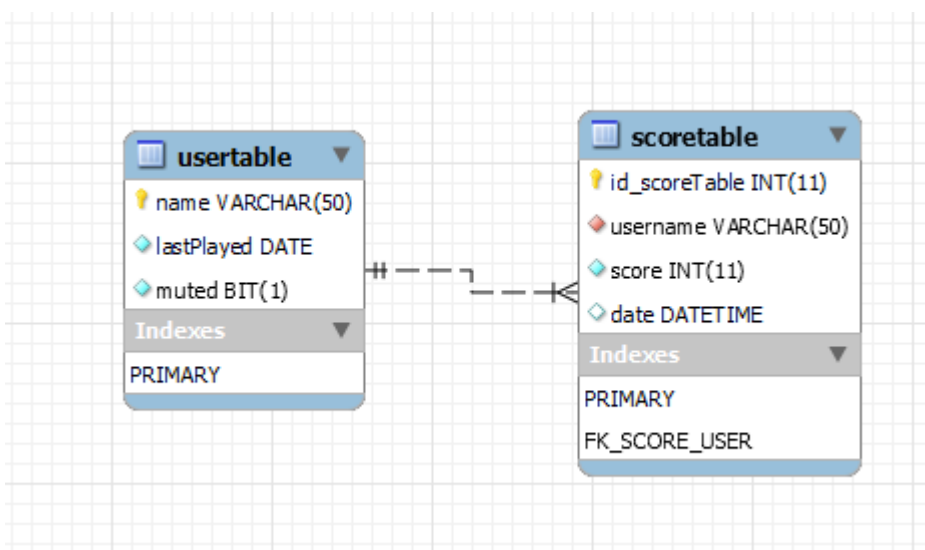
		Subred	Primera IP utilizable	Última IP utilizable	Broadcast
0	0 0 0	20.0.0.0	20.0.0.1	20.31.255.254	20.31.255.255
1	0 0 1	20.32.0.0	20.32.0.1	20.63.255.254	20.63.255.255
2	0 1 0	20.64.0.0	20.64.0.1	20.95.255.254	20.95.255.255
3	0 1 1	20.96.0.0	20.96.0.1	20.127.255.254	20.127.255.255
4	1 0 0	20.128.0.0	20.128.0.1	20.159.255.254	20.159.255.255
5	1 0 1	20.160.0.0	20.160.0.1	20.191.255.254	20.191.255.255
6	1 1 0	20.192.0.0	20.192.0.1	20.223.255.254	20.223.255.255
7	1 1 1	20.224.0.0	20.224.0.1	20.255.255.254	20.255.255.255

3.2 Datos

Dentro de la arquitectura, debe planearse un modelo lógico que permita ingresar y almacenar los datos que se manejarán en nuestra página y que se estarán mostrando al cliente.

Una idea de este modelo se plantea en el siguiente diagrama E-R.

Figura 2 Mapa E-R



Dado que la página es únicamente para jugar, simplificamos la base de datos donde solo hay dos tablas: usuarios y puntuaciones.

3.3 Sistema Operativo

El Sistema Operativo que utilizaremos será Windows Server, el cuál es una distribución de Microsoft para el uso de servidores. Utilizaremos la versión Windows Server 2019 Datacenter, La edición Datacenter está optimizada para la virtualización a gran escala; su licencia permite que un servidor ejecute un número ilimitado de instancias de Windows Server.

Y esto es debido a sus ventajas, las cuales son:

- Fácil de administrar: Este sistema operativo tiene una gran facilidad de uso, lo cual lo convierte en el más comercial del mercado.
- Posee una extensa documentación oficial.
- Menor tiempo de desarrollo: Se logra facilitar mucho el desarrollo de aplicaciones y sistemas que corran sobre servidores Windows.
- Fácil aprendizaje: La curva de aprendizaje para Windows es sumamente inferior a la de otros Sistemas Operativos.
- Soporte del framework de alto rendimiento ASP.NET, cuya nueva versión también es compatible con Linux. Sin embargo, aplicaciones como SharePoint o Exchange siguen siendo exclusivas.
- Para Windows ASP .NET es sin dudas el módulo más importante para los alojamientos Windows, esto se debe a que ASP .NET es un framework altamente demandado a la hora de desarrollo web. Uno de los motivos es que ayuda a agilizar el desarrollo de una aplicación y proporciona código para el dominio de funciones simples.

La Ventaja híbrida de Azure para Windows Server permite ahorrar significativamente en máquinas virtuales de Windows Server. Con la Ventaja híbrida de Azure, solo se tiene que pagar los costes de infraestructura de la máquina virtual, ya que la licencia de Windows Server viene cubierta por el beneficio de Software Assurance.

3.4 Comunicación

3.4.1 Usuario - Página Web

Los usuarios podrán interactuar con la página mediante una computadora, laptop o celular, que tenga acceso a internet, introduciendo la URL que los redireccionará a la página web.

Una vez dentro, tendrán la posibilidad de interactuar con las vistas y hacer uso de las funciones que ofrece la página web.

3.4.2 Página Web - Base de Datos

Dado que utilizamos el patrón de diseño MVC, separamos la lógica y funciones de la base de datos con la interacción del usuario. El modelo se comunica con la base de datos para llevar o sacar información, dependiendo de lo que el controlador le esté solicitando.

3.4.3 Máquina Virtual – Cliente

Le otorgaremos al cliente las llaves necesarias y un nombre de usuario con las que podrá tener una conexión remota a la máquina virtual. Con esto, el cliente puede tener acceso tanto a la máquina virtual como a la base de datos, haciendo posible que el cliente haga las actividades o cambios que desee.

3.5 Módulos

Los módulos se definieron de acuerdo con el patrón de diseño MVC, separándose en 3: Vistas, Controlador y Modelos.

3.5.1 Interfaz Gráfica del Usuario (Vistas)

En cuanto al diseño de la página web se refiere, se pidió que sea desarrollada con HTML5, Javascript, CSS y PHP. También otras herramientas que usamos fueron los Frameworks: JQuery y Bootstrap v5.

3.5.2 Controlador

Este punto entra en juego dependiendo de la acción que hizo el usuario en la vista. Éste convocará "peticiones" de información que se encuentren en la base de datos por medio de los modelos. El controlador hace de intermediario entre la vista y el modelo.

3.5.3 Acceso a Datos (Modelos, DAO)

Ésta es la parte que se comunica directamente con la base de datos. Responde a las peticiones hechas por el controlador y le regresa una respuesta para así, en caso de ser necesario, actualizar la vista. También, por medio de este módulo, se puede editar, eliminar o insertar un registro en una tabla de una base de datos.

3.6 Puertos de Enlace y DNS

El equipo de desarrollo decidió abrir y utilizar los siguientes puertos:

- 80 (HTTP), 443 (HTTPS): Permite a las computadoras fuera de la instancia el conectarse con HTTP Y HTTPS. Si estos puertos no están abiertos, nuestro sitio no puede ser accedido desde fuera de la instancia. Cambiaremos el redireccionamiento a HTTPS cuando se cuente con el certificado necesario.
- 3389 (RDP): Permite el conectarse a la instancia con protocolo de escritorio remoto.
- 1433 (TCP): Es un importante protocolo de red que permite que dos hosts se conecten e intercambien flujos de datos. TCP garantiza la entrega de datos y paquetes en el mismo orden en que fueron enviados.

En cuanto al DNS, por parte del cliente (la máquina virtual), al momento de ser creada se le asigna automáticamente, este es el que se utilizara y no se realizarán cambios.

4. Desarrollo

Las tecnologías que se usarán para el desarrollo de la página web son: HTML, CSS, Bootstrap, JavaScript y JQuery en cuanto a la parte del diseño y PHP para lo referente a la conexión y comunicación de la base de datos, la cual será MySQL. Utilizando el patrón de diseño MVC para separar los datos y principalmente lo que es la lógica de negocio de la aplicación.

5. Apéndices

5.1 Conformidad con Estándares

La aplicación web será creada con los estándares de HTML 5 y CSS3, ECMAScript (ECMA-262) para el manejo de JavaScript y JQuery, SPL para el manejo de PHP y SQL:2003 para el manejo de MySQL.

5.2 Corrección de Errores

Se hará monitoreos cada cierto tiempo para verificar que no haya ningún error en la programación de la aplicación web, también se realizarán múltiples pruebas en determinadas etapas del proyecto para encontrar todos los fallos que haya en ese momento para así corregirlos antes de la publicación de la aplicación web.

5.3 Seguridad

En cuestión de seguridad, la transmisión de información sensible se hará a través de métodos de transferencia POST, encriptados y con tokens para el inicio de sesión, la información no tan relevante se hará a través de métodos GET.