**Grupo Consigue Ventas E.I.R.L.**

***PLAN DE GESTIÓN DE LOS RECURSOS***



**Jefe directo:** Jhoel Fernandéz Alvarado

**Área: Sistemas**

Departamento Estratégico

**LIMA – PERÚ**

**Junio, 2023**

| **CONTROL DE VERSIONES** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Versión** | **Hecha por** | **Revisada por** | **Aprobada por** | **Fecha** | **Motivo** |
| 1.0 | Equipo | Jhoel Fernandéz | Jhoel Fernandéz | 25/05/2023 | Sistema Web |
| 2.0 | Equipo |  |  | 26/05/2023 | Sistema Web |

*PLAN DE GESTIÓN DE LOS RECURSOS*

| **Nombre del proyecto** | **Siglas del Proyecto** |
| --- | --- |
| **Sistema web de asistencia** | **SWA** |

| **Identificación de Los Recursos:** *Métodos para identificar y cuantificar el equipo y los recursos físicos necesarios.* | | |
| --- | --- | --- |
| ***Recursos*** | ***Cantidad*** | |
| ***Ordenadores*** | ***9*** | |
| ***sillas*** | ***9*** | |
| ***escritorios*** | ***9*** | |
| ***Conexión a internet*** | ***9*** | |
| ***software de videoconferencia(google meet)*** | ***1*** | |
| ***aplicación de mensajería(telegram)*** | ***1*** | |
| *Nota: Adjuntar Cuadro de Adquisiciones de Recursos del Proyecto.* | | |
| **Adquisición de Recursos:** *Guías sobre el modo en que se debe adquirir el equipo y los recursos físicos del proyecto.* | | |
| Cada practicante tendrá que tener el equipo necesario para realizar este proyecto | | |
| *Nota: Adjuntar Cuadro de Adquisiciones de Recursos del Proyecto.* | | |
| **Roles y Responsabilidades:** *Nombre del rol, niveles de autoridad, responsabilidad y competencia.* | | |
| 1. **Director de Proyecto:**    1. Niveles de autoridad: Alto    2. Responsabilidades:    3. Planificar, coordinar y supervisar el proyecto en su conjunto.    4. Establecer los objetivos y metas del proyecto.    5. Tomar decisiones estratégicas.    6. Asignar recursos y supervisar el presupuesto.    7. Evaluar el progreso y realizar ajustes según sea necesario.    8. Ser el punto de contacto principal para los interesados y clientes. 2. **Líder de Desarrollo:**    1. Niveles de autoridad: Medio    2. Responsabilidades:    3. Coordinar y gestionar el equipo de desarrollo.    4. Supervisar el avance del proyecto y asegurarse de que se cumplan los plazos.    5. Asignar tareas y responsabilidades al equipo.    6. Comunicarse con el director del proyecto y los miembros del equipo.    7. Resolver problemas y tomar decisiones relacionadas con el desarrollo del software. 3. **Desarrollador Frontend:**    1. Niveles de autoridad: Bajo    2. Responsabilidades:    3. Desarrollar la interfaz de usuario y la parte visual del software.    4. Utilizar tecnologías como HTML, CSS, JavaScript y React.    5. Asegurar la usabilidad y accesibilidad del software.    6. Colaborar con el equipo de diseño para implementar los diseños. 4. **Desarrollador Backend:**    1. Niveles de autoridad: Bajo    2. Responsabilidades:    3. Desarrollar la lógica del software y la gestión de datos.    4. Utilizar tecnologías como PHP con Laravel para crear APIs y gestionar el backend.    5. Implementar la lógica del negocio y la funcionalidad requerida.    6. Garantizar la seguridad y eficiencia del sistema. 5. **Especialista en Bases de Datos:**    1. Niveles de autoridad: Bajo    2. Responsabilidades:    3. Diseñar, implementar y gestionar la base de datos del proyecto.    4. Utilizar MySQL para administrar y asegurar la integridad de los datos.    5. Optimizar el rendimiento de la base de datos.    6. Realizar copias de seguridad y recuperación de datos. | | |
| *Nota: Adjuntar Descripción de Roles.* | | |
| **Organigrama del Proyecto:** *Especificar el organigrama del proyecto.* | | |
| **Director de Proyecto:**   1. **Encargado general del proyecto**   Toma decisiones estratégicas y asegurar el cumplimiento de los objetivos.  Supervisa y coordina todas las actividades del proyecto.  Líder de Desarrollo:   1. **Responsable de la coordinación y supervisión del equipo de desarrollo**   Toma decisiones técnicas y establece las directrices del proyecto.  Desarrolladores Frontend:   1. **Encargados de desarrollar la parte visual y la interfaz de usuario del software**   Utilizan tecnologías como HTML, CSS, JavaScript y React.  Desarrolladores Backend:   1. **Encargados de desarrollar la lógica del software y la gestión de datos**   Utilizan tecnologías como PHP con Laravel para la creación de APIs.   1. **Especialista en Bases de Datos**   Responsable del diseño y la gestión de la base de datos del proyecto  Utilizan MySQL para almacenar y gestionar los datos del software. | | |
| *Nota: Adjuntar Organigrama del Proyecto.* | | |
| **Gestión de Los Recursos del Equipo de Proyecto:** ¿*Cómo definir, proveer personal, administrar y eventualmente liberar los recursos del Equipo de Proyecto?* | | |
|  | |  |
| 1. **Definición de los recursos necesarios:**    1. Identifica los roles y habilidades necesarios para el proyecto, basándote en los requerimientos y objetivos del mismo.    2. Determina la cantidad de recursos necesarios para cada rol.    3. Especifica los conocimientos técnicos, competencias y experiencia requeridos para cada rol. 2. **Provisión de personal:**    1. Anuncia las vacantes disponibles tanto interna como externamente, según corresponda.    2. Evalúa y selecciona a los candidatos más adecuados para cada rol,basándote en sus habilidades y competencias.    3. Realiza entrevistas y pruebas técnicas para evaluar la aptitud de los candidatos.    4. Ofrece las oportunidades de empleo a los candidatos seleccionados. 3. **Administración de recursos:**    1. Asigna a los miembros del equipo a sus roles específicos dentro del proyecto.    2. Proporciona orientación y capacitación necesaria para el desarrollo de las habilidades requeridas.    3. Establece y comunica las expectativas y responsabilidades de cada miembro del equipo.    4. Supervisa y evalúa el desempeño de los recursos del equipo de proyecto.    5. Proporciona retroalimentación regular para el crecimiento y desarrollo del equipo.    6. Proporciona los recursos necesarios, como herramientas, software y equipo de trabajo. 4. **Liberación de recursos:**    1. Revisa periódicamente la necesidad de cada recurso en el proyecto.    2. Identifica si hay cambios en los requerimientos del proyecto que puedan afectar la necesidad de ciertos roles o habilidades.    3. Si un recurso ya no es necesario o si no cumple con las expectativas, considera la posibilidad de liberarlo del proyecto.    4. Realiza una transición adecuada para garantizar una transferencia de responsabilidades suave y efectiva, ya sea asignando tareas a otros miembros del equipo o contratando un reemplazo si es necesario. | |  |

| **Capacitación:** *Estrategias de capacitación para los miembros del equipo.* |
| --- |
| 1. **Identificar las necesidades de capacitación:** Realiza una evaluación de las habilidades y conocimientos actuales de tu equipo. Identifica las áreas en las que necesitan mejorar o adquirir nuevos conocimientos relacionados con su rol en el proyecto. 2. **Establecer metas de capacitación:** Define metas claras y específicas para la capacitación de cada miembro del equipo. Estas metas deben estar alineadas con los objetivos del proyecto y las necesidades individuales de desarrollo. 3. **Seleccionar cursos relevantes en Udemy:** Explora la amplia gama de cursos disponibles en Udemy y elige aquellos que sean relevantes para las necesidades de capacitación identificadas. Asegúrate de que los cursos seleccionados cubren los temas requeridos y estén actualizados. 4. **Planificar el tiempo y recursos necesarios:** Determina cuánto tiempo se requerirá para completar cada curso y asigna los recursos necesarios, como acceso a la plataforma Udemy, licencias o materiales adicionales que los miembros del equipo puedan necesitar. 5. **Fomentar la participación activa:** Anima a los miembros del equipo a aprovechar al máximo los cursos en Udemy. Motívelos a participar activamente en las actividades del curso, como realizar ejercicios prácticos, participar en discusiones y completar los cuestionarios o exámenes. 6. **Monitorear el progreso y brindar apoyo:** Realiza un seguimiento regular del progreso de cada miembro del equipo en sus cursos. Ofrece apoyo y asistencia adicional en caso de que encuentren dificultades o necesiten aclarar concepto**s** 7. **Aplicar los conocimientos adquiridos:** Al finalizar cada curso, fomenta la aplicación práctica de los conocimientos adquiridos en el contexto del proyecto. Proporciona oportunidades para que los miembros del equipo apliquen lo aprendido y compartan sus experiencias con el resto del equipo. 8. **Retroalimentación y evaluación:** Solicita retroalimentación a los miembros del equipo sobre la calidad y utilidad de los cursos en Udemy. Realiza evaluaciones periódicas para medir el impacto de la capacitación en el desempeño y el logro de los objetivos del proyecto. |
| **Desarrollo del Equipo:** *Métodos para desarrollar el equipo.* |
| 1. **Formación inicial en Scrum:** Proporciona una formación inicial en Scrum a todos los miembros del equipo. Esto les ayudará a comprender los principios y valores fundamentales de Scrum, así como los roles, eventos y artefactos clave. 2. **Roles y responsabilidades claras:** Cada miembro del equipo entienda su rol y responsabilidades dentro del marco de Scrum. Esto incluye el Scrum Master, el Product Owner y los miembros del Equipo de Desarrollo. 3. **Prácticas de colaboración:** Fomenta la colaboración entre los miembros del equipo. Utiliza técnicas como el emparejamiento de programadores, revisiones de código y sesiones de lluvia de ideas para promover el intercambio de conocimientos y la resolución conjunta de problemas. 4. **Establecimiento de metas y objetivos:** Ayuda al equipo a establecer metas y objetivos claros y alcanzables. Estas metas pueden ser tanto a nivel individual como colectivo, y deben estar alineadas con los objetivos del proyecto y las necesidades del cliente. 5. **Retroalimentación continua:** Implementa prácticas regulares de retroalimentación, como las retrospectivas del sprint, donde el equipo puede reflexionar sobre su desempeño, identificar áreas de mejora y definir acciones correctivas. 6. **Desarrollo de habilidades:** Identifica las habilidades y conocimientos necesarios para el éxito del equipo y brinda oportunidades de desarrollo. Esto puede incluir la participación en cursos, talleres, conferencias o la asignación de tareas que desafíen y amplíen las habilidades existentes. 7. **Empoderamiento del equipo:** Fomenta un entorno en el que los miembros del equipo se sientan empoderados y puedan tomar decisiones. Promueve la autogestión y la toma de responsabilidad en la planificación y ejecución del trabajo. 8. **Comunicación efectiva:** Establece canales de comunicación claros y efectivos dentro del equipo. Esto incluye reuniones diarias de Scrum (stand-ups), donde se comparten actualizaciones y se identifican posibles obstáculos, y la utilización de herramientas de colaboración en línea. 9. **Aprendizaje continuo:** Fomenta una cultura de aprendizaje continuo dentro del equipo. Esto implica alentar a los miembros del equipo a buscar nuevas oportunidades de crecimiento, compartir conocimientos y aprender de las experiencias pasadas. 10. **Mejora continua:** Promueve la mejora continua del equipo. Esto puede incluir la experimentación con nuevas técnicas o prácticas, la implementación de cambios basados en retroalimentación y la adaptación constante del proceso de desarrollo. |
| **Control de Recursos:** *los métodos para asegurar que los recursos físicos adecuados estén disponibles cuando sean necesarios y que la adquisición de recursos físicos sea optimizada para las necesidades del proyecto. incluir información sobre la gestión de inventario, equipos y suministros.* |
| 1. **Evaluación de necesidades:** Realiza una evaluación exhaustiva de los requisitos de software del proyecto. Identifica las funcionalidades, características y tecnologías necesarias para el desarrollo del software. 2. **Selección de herramientas:** Basándote en los requisitos del proyecto, investiga y selecciona las herramientas de software más adecuadas. Considera factores como la compatibilidad, la escalabilidad, la seguridad y la facilidad de uso. 3. **Adquisición de licencias:** Si el software requiere licencias, asegúrate de adquirir las licencias adecuadas para el número de usuarios y las funcionalidades necesarias. Mantén un registro de las licencias adquiridas y su período de validez. 4. **Gestión de versiones:** Establece un sistema de control de versiones para gestionar las diferentes versiones del software. Utiliza herramientas como Git para controlar los cambios, fusionar contribuciones y garantizar la integridad del código fuente. 5. **Pruebas y validación:** Realiza pruebas exhaustivas del software para garantizar su funcionalidad y rendimiento. Esto puede incluir pruebas unitarias, pruebas de integración y pruebas de aceptación para verificar que el software cumple con los requisitos establecidos. 6. **Mantenimiento y actualizaciones:** Establece un plan de mantenimiento para asegurar que el software se mantenga actualizado y funcional. Esto puede incluir la instalación de parches de seguridad, actualizaciones de características y corrección de errores. 7. **Gestión de licencias y renovaciones:** Mantén un seguimiento de las licencias de software y su período de vigencia. Realiza las renovaciones necesarias antes de que expiren para garantizar la continuidad del uso del software. Incluir la experimentación con nuevas técnicas o prácticas, la implementación de cambios basados en retroalimentación y la adaptación constante del proceso de desarrollo. |
| **Plan de Reconocimiento:** ¿*Qué reconocimiento y recompensa se dará a los miembros del equipo?* |
| 1. **Certificado de prácticas satisfactorias:** Al finalizar el proyecto, se otorgará a cada miembro del equipo un certificado que valide su participación y desempeño satisfactorio en el proyecto. 2. **Conocimiento del entorno del software:** Se brindará a los miembros del equipo la oportunidad de adquirir un conocimiento profundo del entorno de software utilizado en el proyecto. Esto les permitirá estar preparados para futuros trabajos y proyectos relacionados con el mismo software. 3. **Capacitación con cursos:** Se proporcionarán cursos y capacitaciones adicionales para potenciar las habilidades y conocimientos de los miembros del equipo. Estos cursos pueden estar relacionados con tecnologías específicas utilizadas en el proyecto, metodologías ágiles, desarrollo de software, gestión de proyectos, entre otros. |