|  |  |
| --- | --- |
| **Detalle de Diseño Nº** | 01 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Elaborado Por:** | Eysenck Gómez Orihuela | | | |
| **Descripción General:** | Este documento está orientado a servir de guía de desarrollo de la funcionalidad:  Gestión de comentarios:   * Esta funcionalidad tiene como finalidad registrar todos los comentarios hechos por los usuarios de tipo empleado así como los usuarios del tipo administrador A y B. * Esta funcionalidad también debe permitir que los administradores de tipo A puedan aprobar los comentarios de los empleados para que recién estos puedan ser visibles para los administradores de tipo B. | | | |
| **Referencias:** | **Requerimientos:** | * SGT-ECU-GC.dox Especificación de caso de uso Gestionar Comentario | | |
| **Diseño de Interfaz de Usuario:** | * SGT-LRFNF.docx Lista de requisitos funcionales y no funcionales | | |
| **Diseño de Arquitectura:** | * SGT-LRFNF.docx Lista de requisitos funcionales y no funcionales | | |
| **Clases Requeridas:** | **Nombre de la Clase** | **Stereotipo** | | **Acción** |
| Comentario | Model | | Crear |
| User | Model | | Listar |
| Perfil | Model | | Listar |
| Solicitud | Model | | Listar |
| **HTML** | **Nombre de la plantilla (HTML)** | | **Acción** | |
| Ver\_comentarios.html | | Listar | |
| **Vistas (Views)** | **Nombre de la vista** | | **Acción** | |
| Agregar\_comentario | | Crear | |
| Aprobar\_comentario | | Actualizar | |
| Detalles\_solicitud | | Listar | |
| **Tablas** | **Nombre de Tabla** | | **Tipo Objeto** | |
| Appsistema\_comentario (select, insert, update) | | Tabla | |
| Auth\_user (select) | | Tabla | |
| Appsistema\_solicitud (select) | | Tabla | |
| **Detalle de Implementación a Nivel de Clases:** | **appsistema.models.Comentario**   * Clase Python de persistencia utilizada para representar la información de la tabla Comentario y facilitad la programación en un ambiente Object Relational Mapping. Mediante la configuración de los respecitvos archivos descriptores, será utilizada bajo el estereotipo de Model. Esto permitirá realizar operaciones transparentes de Inserción, Actualización y Consulta sobre dicha tabla, sin necesidad de hacer consultas nativas sobre la base de datos. * **Acción:** Crear   **django.cotrib.auth.models.User**   * Clase Python (Model) para la persistencia de los Usuarios del sistema. * **Acción:** Crear   **appsistema.models.Solicitud**   * Clase Python (Model) para la persistencia de las Solicitudes hechas en el sistema. | | | |
| **Detalle de Implementación a Nivel de Plantillas HTML:** | **Ver\_comentarios.html**   * **Ruta:** templates/ver\_comentario.html * **Acción:** Listar * **Descripción:** Objeto Plantilla utilizado para la Interfaz Gráfica de Usuario GUI de la gestión de comentarios. Utiliza los tags propios del sistema de plantillas de Django. | | | |
| **Detalle de Implementación a Nivel de Tablas:** | **Tabla Appsistema\_comentario**  Tabla que almacena los ingresos y descuentos realizados a los empleados para el cálculo de su planilla en un período dado:  CREATE TABLE `appsistema\_comentario` (  `id` integer AUTO\_INCREMENT NOT NULL PRIMARY KEY,  `emisor\_id` integer NOT NULL,  `solicitud\_id` integer NOT NULL,  `texto` longtext NOT NULL,  `fecha` datetime NOT NULL,  `aprobado` bool NOT NULL  );  ALTER TABLE `appsistema\_comentario` ADD CONSTRAINT `emisor\_id\_refs\_id\_6fa937cc` FOREIGN KEY (`emisor\_id`) REFERENCES `auth\_user` (`id`);  ALTER TABLE `appsistema\_comentario` ADD CONSTRAINT `solicitud\_id\_refs\_id\_5bbf1ee1` FOREIGN KEY (`solicitud\_id`) REFERENCES `appsistema\_solicitud` (`id`);  CREATE INDEX `appsistema\_comentario\_3e4e6543` ON `appsistema\_comentario` (`emisor\_id`);  CREATE INDEX `appsistema\_comentario\_579b8b39` ON `appsistema\_comentario` (`solicitud\_id`);  COMMIT;  Es una tabla autoincrementada. Tiene una llave foránea a solicitud vía el campo solicitud\_id y elconstraint emisor\_id\_refs\_id\_6fa937cc. | | | |