

ESQUEMA de apartados para la documentación del Proyecto.

Portada + Índice

Cada página debe llevar encabezado, pie de página, y números de páginas.

1. Pequeña introducción/ descripción. (½ Página)

- De donde surge la necesidad

La Orquesta Filarmónica de Gran Canaria necesita una app para organizar sus proyectos.

- Para que empresa se desarrolla

La Orquesta Filarmónica de Gran Canaria.

- Cual es la idea resumida (máximo dos párrafos) del desarrollo

La propuesta es una app donde los músicos puedan ver su “calendario” y los administradores puedan editarlo a través del backend.

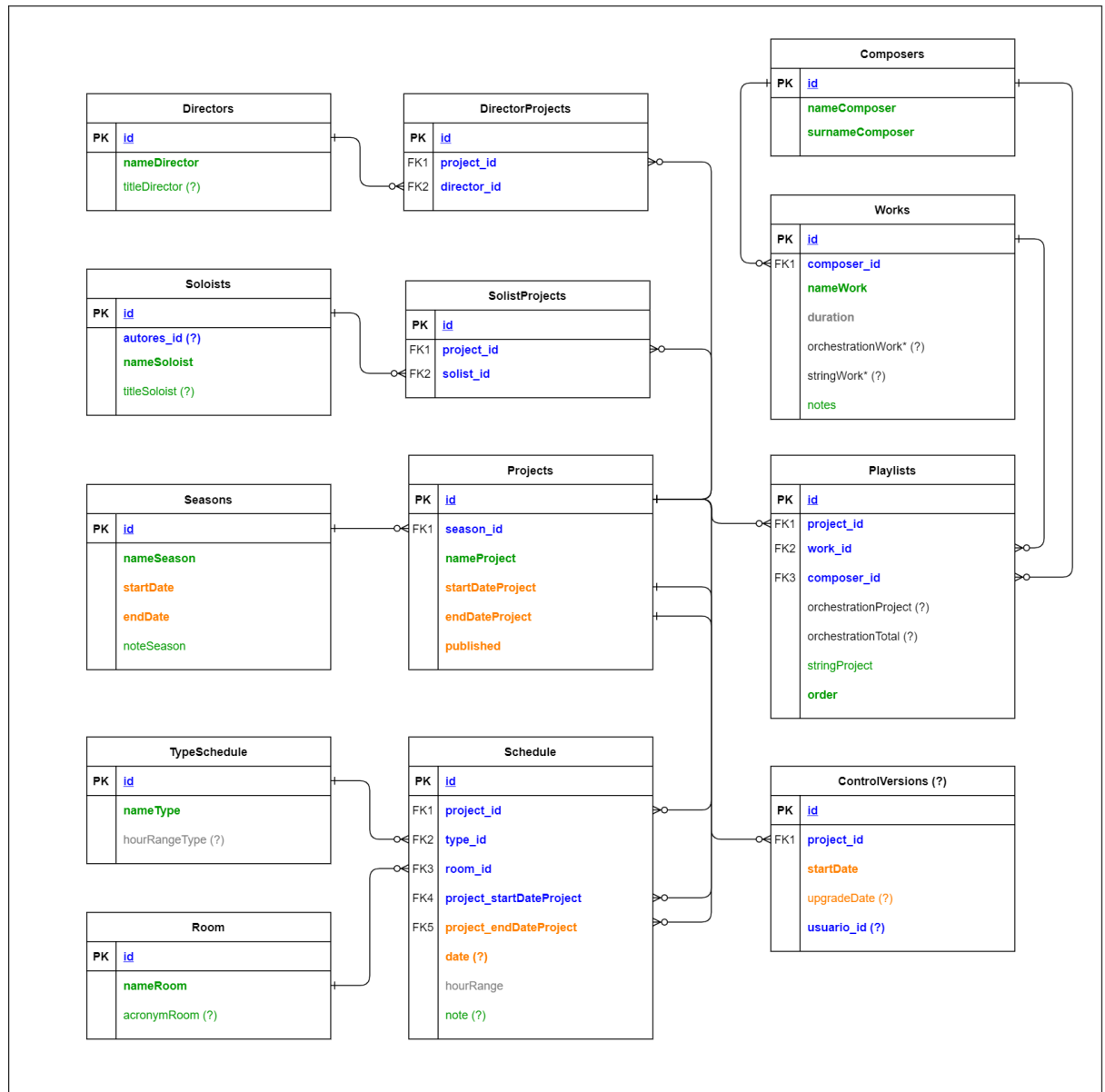
- Otros

2. Diagramas, su justificación. Explicación del “Modelo de Datos”

- Ver apartado [Documentación modelo de datos](#) [Archivo](#) del módulo "Acceso a Datos/ Recursos Proyecto Integración AED. (... de clases,

entidad/relación, ... otros)

Leyenda	
Negrita	NOT NULL
Verde	String
Azul	int
Naranja	Date
Gris	Time
Duda	(?)



Entitys:

*every entity has an id which is primary key, int, not null and with auto-increment.

-añadir si los deletes todo van a ser los mismos-

Directors: nameDirector (String, NOT NULL) director's name and surname, titleDirector (String) (?).

DirectorsProjects: project_id (foreign key, int, NOT NULL) id from Projects entity, soloist_id (foreign key, NOT NULL), id from Soloists entity.

Soloists: autores_id (int, NOT NULL) (?), nameSoloist (String, NOT NULL) soloist's name and surname, titleDirector (String) (?).

SoloistProjects: project_id (foreign key, NOT NULL) id from Projects entity, director_id (foreign key, NOT NULL), id from Directors entity.

Seasons: nameSeason (String, NOT NULL) name of the season, startDate (Date) the date that the season starts, endDate (Date) the date that the season ends, noteSeason (String) notes, ideas or description, that can be added about the season.

Projects: season_id (foreign key, int, NOT NULL) id from Seasons entity, nameProject (String NOT NULL) name of the project, startDateProject (Date, NOT NULL) the date that the project starts, endDateProject (Date, NOT NULL) the date that the project ends, published (Date, NOT NULL) if the project is published or not.

Schedule: project_id (int, NOT NULL), type_id (int, NOT NULL) id from the TypeSchedule entity, room_id (int, NOT NULL) id from the entity Room, date (Date, NOT NULL) date's schedule which is determined in a range between the start day of a project and its end, hourRange (Time) hour of the schedule, note (String) (?).

TypeSchedule: nameType (String, NOT NULL) if it is an essay or a performance from the TypeSchedule entity, hourRangeType (Time) (?).

Room: id (int, NOT NULL, auto-increment), nameRoom (String, NOT NULL) name of the room, acronymRoom (String) (?)

Composers: nameComposer (String, NOT NULL) Composer's name, surnameComposer (String, NOT NULL) Composer's surname.

Works: composer_id (foreign key, int, NOT NULL) id from the Composer entity, nameWork (String, NOT NULL) title of the work, duration (Time, NOT NULL) how much it last, orchestrationWork (?), stringWork (?), notes (String) notes that can be added, like ideas or thoughts.

Playlists: project_id (foreign key, int, NOT NULL) id from the Project entity, work_id (foreign key, int, NOT NULL) id from the Work entity, composer_id (foreign key, int, NOT NULL) id from the Composer entity, orchestrationProject (?), orchestrationTotal (?), stringProject (String), order (String, NOT NULL) order of the works inside the playlist.

ControlVersions (?): project_id (foreign key, int, NOT NULL) id from the Project entity, startDate (Date, NOT NULL), upgradeDate (?), usuario_id (int NOT NULL) (?)

Base de datos:

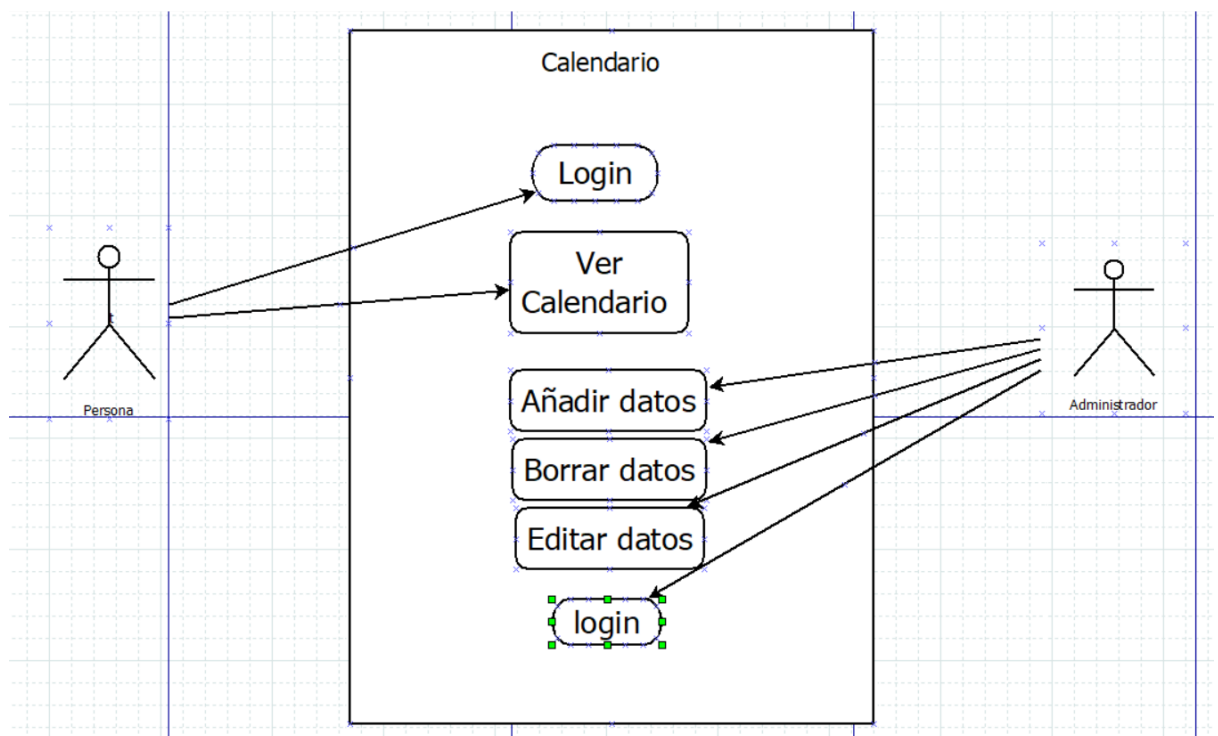
SQL: (se hizo a través de migración y ORM de Laravel, es decir, así que no se ejecutaron secuencias sql manualmente, se podría poner aquí el código de cada modelo).

3. Requisitos de Usuario.

- Su objetivo principal es comenzar a definir el sistema software (la solución) que satisfaga las necesidades de negocio de clientes y usuarios (el problema). Para ello, se definen los **requisitos generales** y se comienzan a especificar los casos de uso del sistema en su versión inicial. Conforme avance el proceso de ingeniería de requisitos, los requisitos generales se irán detallando en requisitos más **específicos** y los casos de uso que se considere oportuno irán evolucionando hacia su forma detallada.

Salida / lo que se pide: Requisitos generales del sistema, principal producto resultante de esta tarea junto con la versión inicial de los casos de uso. Deben especificar las características principales del sistema a desarrollar.

4. Casos de Usos:



Los casos de uso son del usuario promedio (los músicos) son: ver los proyectos y calendarios de estos, es decir, consultar toda la información que le corresponde. También debe poder logearse para ver aquello que le corresponde.

El administrador podrá añadir, borrar, editar y ver TODOS los datos, además de hacer login para otorgarle los permisos correspondientes para ello.

5. Describir el funcionamiento del sistema y especificaciones técnicas para la aplicación del servidor y las APP's móvil y web.
 - Añadir en esta descripción algún elemento **visual** que explique el esquema y las relaciones (por ejemplo un mapa conceptual)

6. Interfaces:

1. Diseño Inicial: Mokups y prototipado

2. Usabilidad

- Lista de aspectos de la rúbrica de Usabilidad que vas a tener en cuenta.
- Para cada aspecto: Justificar porqué lo vas a diseñar así, qué vas a tener en cuenta.
- Cual es el **proceso** seguido para estudiar la Usabilidad, previo al desarrollo
- Una vez se ha desarrollado la aplicación: Añadir la lista con descripciones e imagen demostrativa de nuestra app, en la que se aplican los aspectos de usabilidad de **cada** aspecto.

7. Manuales

1. Manual de instalación para el desarrollador (tanto servidor como cliente)
2. Manual de instalación para los técnicos que van a instalar la aplicación al cliente que compre la Aplicación.
3. Manual de usuario
4. Ayuda a usuario dentro de la app

8. Pila tecnológica

9. Comparación de tecnologías (½ Página)

10. Repositorios

11. Planificación: Explica como te has organizado y planificado (Media Página)

12. Conclusiones, opiniones, reflexiones (auténticas y no lo hagan por rellenar) (½ Página)

13. Enlaces y referencias

ANEXOS

- Recomendamos completar este esquema con la plantilla de [Plantilla funciones del sistema \(modelo ERS\) Archivo](#) que encontrarás en el módulo de "Acceso a Datos".

- En ese documento puedes resolver dudas sobre "requisitos de usuario" y otros aspectos que puedes encontrar en cada apartado.