openmusic-api

Struktur file dan kode

>migrations

>1750589174371\_create-table-albums.js

exports.shorthands = undefined;

exports.up = (pgm) => {

  pgm.createTable('albums', {

    id: {

      type: 'VARCHAR(50)',

      primaryKey: true,

    },

    name: {

      type: 'TEXT',

      notNull: true,

    },

    year: {

      type: 'INTEGER',

      notNull: true,

    },

  });

};

exports.down = (pgm) => {

  pgm.dropTable('albums');

};

>1750589194823\_create-table-songs.js

exports.shorthands = undefined;

exports.up = (pgm) => {

  pgm.createTable('songs', {

    id: {

      type: 'VARCHAR(50)',

      primaryKey: true,

    },

    title: {

      type: 'TEXT',

      notNull: true,

    },

    year: {

      type: 'INTEGER',

      notNull: true,

    },

    genre: {

      type: 'TEXT',

      notNull: true,

    },

    performer: {

      type: 'TEXT',

      notNull: true,

    },

    duration: {

      type: 'INTEGER',

      allowNull: true,

    },

    album\_id: {

      type: 'VARCHAR(50)',

      references: '"albums"',

      onDelete: 'CASCADE',

      allowNull: true,

    },

  });

};

exports.down = (pgm) => {

  pgm.dropTable('songs');

};

>1750589208549\_create-table-users.js

exports.shorthands = undefined;

exports.up = (pgm) => {

  pgm.createTable('users', {

    id: {

      type: 'TEXT',

      primaryKey: true,

    },

    username: {

      type: 'TEXT',

      notNull: true,

      unique: true,

    },

    password: {

      type: 'TEXT',

      notNull: true,

    },

    fullname: {

      type: 'TEXT',

      notNull: true,

    },

  });

};

exports.down = (pgm) => {

  pgm.dropTable('users');

};

>1750729862753\_add-playlists-table.js

exports.up = (pgm) => {

  pgm.createTable('playlists', {

    id: { type: 'text', primaryKey: true },

    name: { type: 'text', notNull: true },

    owner: { type: 'text', notNull: true, references: 'users(id)', onDelete: 'cascade' },

  });

  pgm.createTable('playlist\_songs', {

    id: { type: 'text', primaryKey: true },

    playlist\_id: { type: 'text', notNull: true, references: 'playlists(id)', onDelete: 'cascade' },

    song\_id: { type: 'text', notNull: true, references: 'songs(id)', onDelete: 'cascade' },

  });

};

exports.down = (pgm) => {

  pgm.dropTable('playlist\_songs');

  pgm.dropTable('playlists');

};

>1751517360219\_create-table-collaborations.js

exports.up = (pgm) => {

  pgm.createTable('collaborations', {

    id: {

      type: 'VARCHAR(50)',

      primaryKey: true,

    },

    playlist\_id: {

      type: 'VARCHAR(50)',

      notNull: true,

      references: '"playlists"',

      onDelete: 'cascade',

    },

    user\_id: {

      type: 'VARCHAR(50)',

      notNull: true,

      references: '"users"',

      onDelete: 'cascade',

    },

    created\_at: {

      type: 'TIMESTAMP',

      notNull: true,

      default: pgm.func('current\_timestamp'),

    },

    updated\_at: {

      type: 'TIMESTAMP',

    },

  });

  pgm.createIndex('collaborations', 'playlist\_id');

  pgm.createIndex('collaborations', 'user\_id');

  pgm.createIndex('collaborations', ['playlist\_id', 'user\_id'], { unique: true });

};

exports.down = (pgm) => {

  pgm.dropTable('collaborations');

};

>1751521167220\_create-table-playlist-activities.js.js

exports.up = (pgm) => {

  pgm.createTable('playlist\_song\_activities', {

    id: {

      type: 'VARCHAR(50)',

      primaryKey: true,

    },

    playlist\_id: {

      type: 'VARCHAR(50)',

      notNull: true,

      references: '"playlists"',

      onDelete: 'CASCADE',

    },

    song\_id: {

      type: 'VARCHAR(50)',

      notNull: true,

      references: '"songs"',

      onDelete: 'CASCADE',

    },

    user\_id: {

      type: 'VARCHAR(50)',

      notNull: true,

      references: '"users"',

      onDelete: 'CASCADE',

    },

    action: {

      type: 'VARCHAR(50)',

      notNull: true,

    },

    time: {

      type: 'TIMESTAMP',

      notNull: true,

      default: pgm.func('current\_timestamp'),

    },

  });

  pgm.createIndex('playlist\_song\_activities', 'playlist\_id');

};

exports.down = (pgm) => {

  pgm.dropTable('playlist\_song\_activities');

};

>src

>api

>albums

>handlerAlbums.js

class AlbumsHandler {

  constructor(service, validator) {

    this.\_service = service;

    this.\_validator = validator;

    this.postAlbumHandler = this.postAlbumHandler.bind(this);

    this.getAlbumsHandler = this.getAlbumsHandler.bind(this);

    this.getAlbumByIdHandler = this.getAlbumByIdHandler.bind(this);

    this.putAlbumByIdHandler = this.putAlbumByIdHandler.bind(this);

    this.deleteAlbumByIdHandler = this.deleteAlbumByIdHandler.bind(this);

  }

  async postAlbumHandler(request, h) {

    try {

      this.\_validator.validateAlbumPayload(request.payload);

      const { name, year } = request.payload;

      const albumId = await this.\_service.addAlbum({ name, year });

      return h.response({

        status: 'success',

        data: { albumId },

      }).code(201);

    } catch (error) {

      console.error('Error in postAlbumHandler:', error);

      throw error; // biarkan onPreResponse yang menangani

    }

  }

  async getAlbumsHandler() {

    try {

      const albums = await this.\_service.getAlbums();

      return {

        status: 'success',

        data: { albums },

      };

    } catch (error) {

      console.error('Error in getAlbumsHandler:', error);

      throw error;

    }

  }

  async getAlbumByIdHandler(request) {

    try {

      const { id } = request.params;

      const album = await this.\_service.getAlbumById(id);

      return {

        status: 'success',

        data: { album },

      };

    } catch (error) {

      console.error('Error in getAlbumByIdHandler:', error);

      throw error;

    }

  }

  async putAlbumByIdHandler(request) {

    try {

      this.\_validator.validateAlbumPayload(request.payload);

      const { id } = request.params;

      const { name, year } = request.payload;

      await this.\_service.editAlbumById(id, { name, year });

      return {

        status: 'success',

        message: 'Album berhasil diperbarui',

      };

    } catch (error) {

      console.error('Error in putAlbumByIdHandler:', error);

      throw error;

    }

  }

  async deleteAlbumByIdHandler(request) {

    try {

      const { id } = request.params;

      await this.\_service.deleteAlbumById(id);

      return {

        status: 'success',

        message: 'Album berhasil dihapus',

      };

    } catch (error) {

      console.error('Error in deleteAlbumByIdHandler:', error);

      throw error;

    }

  }

}

module.exports = AlbumsHandler;

>routesAlbums.js

const routes = (handler) => [

  {

    method: 'POST',

    path: '/albums',

    handler: handler.postAlbumHandler,

  },

  {

    method: 'GET',

    path: '/albums',

    handler: handler.getAlbumsHandler,

  },

  {

    method: 'GET',

    path: '/albums/{id}',

    handler: handler.getAlbumByIdHandler,

  },

  {

    method: 'PUT',

    path: '/albums/{id}',

    handler: handler.putAlbumByIdHandler,

  },

  {

    method: 'DELETE',

    path: '/albums/{id}',

    handler: handler.deleteAlbumByIdHandler,

  },

];

module.exports = routes;

>validatorAlbums.js

const Joi = require('joi');

const InvariantError = require('../../exceptions/InvariantError');

const AlbumPayloadSchema = Joi.object({

  name: Joi.string().required(),

  year: Joi.number().required(),

});

const AlbumsValidator = {

  validateAlbumPayload: (payload) => {

    const validationResult = AlbumPayloadSchema.validate(payload);

    if (validationResult.error) {

      throw new InvariantError(validationResult.error.message);

    }

  },

};

module.exports = AlbumsValidator;

>authentications

>handler.js

const autoBind = require('auto-bind');

class AuthenticationsHandler {

  constructor(authenticationsService, usersService, tokenManager, validator) {

    this.\_authenticationsService = authenticationsService;

    this.\_usersService = usersService;

    this.\_tokenManager = tokenManager;

    this.\_validator = validator;

    autoBind(this);

  }

  async postAuthenticationHandler(request, h) {

    this.\_validator.validateAuthPayload(request.payload);

    const { username, password } = request.payload;

    const id = await this.\_usersService.verifyUserCredential(username, password);

    const accessToken = this.\_tokenManager.generateAccessToken({ id });

    const refreshToken = this.\_tokenManager.generateRefreshToken({ id });

    await this.\_authenticationsService.addToken(refreshToken);

    const response = h.response({

      status: 'success',

      data: {

        accessToken,

        refreshToken,

      },

    });

    response.code(201);

    return response;

  }

  async putAuthenticationHandler(request) {

    this.\_validator.validateRefreshTokenPayload(request.payload);

    const { refreshToken } = request.payload;

    await this.\_authenticationsService.verifyRefreshToken(refreshToken);

    const { id } = this.\_tokenManager.verifyRefreshToken(refreshToken);

    const accessToken = this.\_tokenManager.generateAccessToken({ id });

    return {

      status: 'success',

      data: {

        accessToken,

      },

    };

  }

  async deleteAuthenticationHandler(request) {

    this.\_validator.validateRefreshTokenPayload(request.payload);

    const { refreshToken } = request.payload;

    await this.\_authenticationsService.deleteToken(refreshToken);

    return {

      status: 'success',

      message: 'Refresh token berhasil dihapus',

    };

  }

}

module.exports = AuthenticationsHandler;

>index.js

const AuthenticationsHandler = require('./handler');

const routes = require('./routes');

module.exports = {

  name: 'authentications',

  version: '1.0.0',

  register: async (server, {

    authenticationsService,

    usersService,

    tokenManager,

    authenticationsValidator,

  }) => {

    const authenticationsHandler = new AuthenticationsHandler(

      authenticationsService,

      usersService,

      tokenManager,

      authenticationsValidator,

    );

    server.route(routes(authenticationsHandler));

  },

};

>routes.js

const routes = (handler) => [

  {

    method: 'POST',

    path: '/authentications',

    handler: handler.postAuthenticationHandler,

  },

  {

    method: 'PUT',

    path: '/authentications',

    handler: handler.putAuthenticationHandler,

  },

  {

    method: 'DELETE',

    path: '/authentications',

    handler: handler.deleteAuthenticationHandler,

  },

];

module.exports = routes;

>validatorAuthentications.js

const Joi = require('joi');

const InvariantError = require('../../exceptions/InvariantError');

const PostAuthenticationPayloadSchema = Joi.object({

  username: Joi.string().required(),

  password: Joi.string().required(),

});

const PutAuthenticationPayloadSchema = Joi.object({

  refreshToken: Joi.string().required(),

});

const DeleteAuthenticationPayloadSchema = Joi.object({

  refreshToken: Joi.string().required(),

});

const AuthenticationsValidator = {

  validateAuthPayload: (payload) => {

    const result = PostAuthenticationPayloadSchema.validate(payload);

    if (result.error) {

      throw new InvariantError(result.error.message);

    }

  },

  validateRefreshTokenPayload: (payload) => {

    const result = PutAuthenticationPayloadSchema.validate(payload);

    if (result.error) {

      throw new InvariantError(result.error.message);

    }

  },

};

module.exports = AuthenticationsValidator;

>collaborations

>handler.js

const autoBind = require('auto-bind');

class CollaborationsHandler {

  constructor(collaborationsService, playlistsService, validator) {

    this.\_collaborationsService = collaborationsService;

    this.\_playlistsService = playlistsService;

    this.\_validator = validator;

    autoBind(this);

  }

  async postCollaborationHandler(request, h) {

    this.\_validator.validateCollaborationPayload(request.payload);

    const { playlistId, userId } = request.payload;

    const { id: credentialId } = request.auth.credentials;

    await this.\_playlistsService.verifyPlaylistOwner(playlistId, credentialId);

    const collaborationId = await this.\_collaborationsService.addCollaboration(playlistId, userId);

    const response = h.response({

      status: 'success',

      message: 'Kolaborasi berhasil ditambahkan',

      data: {

        collaborationId,

      },

    });

    response.code(201);

    return response;

  }

  async deleteCollaborationHandler(request) {

    this.\_validator.validateCollaborationPayload(request.payload);

    const { playlistId, userId } = request.payload;

    const { id: credentialId } = request.auth.credentials;

    await this.\_playlistsService.verifyPlaylistOwner(playlistId, credentialId);

    await this.\_collaborationsService.deleteCollaboration(playlistId, userId);

    return {

      status: 'success',

      message: 'Kolaborasi berhasil dihapus',

    };

  }

}

module.exports = CollaborationsHandler;

>index.js

const CollaborationsHandler = require('./handler');

const routes = require('./routes');

module.exports = {

  name: 'collaborations',

  version: '1.0.0',

  register: async (server, { collaborationsService, playlistsService, validator }) => {

    const collaborationsHandler = new CollaborationsHandler(

      collaborationsService,

      playlistsService,

      validator

    );

    server.route(routes(collaborationsHandler));

  },

};

>routes.js

const routes = (handler) => [

  {

    method: 'POST',

    path: '/collaborations',

    handler: handler.postCollaborationHandler,

    options: {

      auth: 'openmusic\_jwt',

    },

  },

  {

    method: 'DELETE',

    path: '/collaborations',

    handler: handler.deleteCollaborationHandler,

    options: {

      auth: 'openmusic\_jwt',

    },

  },

];

module.exports = routes;

>validator.js

const Joi = require('joi');

const InvariantError = require('../../exceptions/InvariantError');

const CollaborationPayloadSchema = Joi.object({

  playlistId: Joi.string().required(),

  userId: Joi.string().required(),

});

const CollaborationsValidator = {

  validateCollaborationPayload: (payload) => {

    const validationResult = CollaborationPayloadSchema.validate(payload);

    if (validationResult.error) {

      throw new InvariantError(validationResult.error.message);

    }

  },

};

module.exports = CollaborationsValidator;

>playlists

>index.js

const PlaylistsHandler = require('./PlaylistsHandler');

const routes = require('./routes');

module.exports = {

  name: 'playlists',

  version: '1.0.0',

  register: async (

    server,

    { playlistsService, playlistsValidator, playlistSongsService } // ⬅️ ditambahkan

  ) => {

    const playlistsHandler = new PlaylistsHandler(

      playlistsService,

      playlistsValidator,

      playlistSongsService // ⬅️ disuntikkan ke handler

    );

    server.route(routes(playlistsHandler));

  },

};

>PlaylistsHandler.js

const autoBind = require('auto-bind');

class PlaylistsHandler {

  constructor(service, validator, playlistSongsService) {

    this.\_service = service;

    this.\_validator = validator;

    this.\_playlistSongsService = playlistSongsService;

    autoBind(this);

  }

  async postPlaylistHandler(request, h) {

    try {

      this.\_validator.validatePlaylistPayload(request.payload);

      const { name } = request.payload;

      const { id: owner } = request.auth.credentials;

      const playlistId = await this.\_service.addPlaylist({ name, owner });

      const response = h.response({

        status: 'success',

        data: { playlistId },

      });

      response.code(201);

      return response;

    } catch (error) {

      console.error('Error in postPlaylistHandler:', error);

      throw error;

    }

  }

  async getPlaylistsHandler(request) {

    try {

      const { id: owner } = request.auth.credentials;

      const playlists = await this.\_service.getPlaylists(owner);

      return {

        status: 'success',

        data: { playlists },

      };

    } catch (error) {

      console.error('Error in getPlaylistsHandler:', error);

      throw error;

    }

  }

  async deletePlaylistHandler(request) {

    try {

      const { id } = request.params;

      const { id: owner } = request.auth.credentials;

      await this.\_service.verifyPlaylistOwner(id, owner);

      await this.\_service.deletePlaylist(id);

      return {

        status: 'success',

        message: 'Playlist berhasil dihapus',

      };

    } catch (error) {

      console.error('Error in deletePlaylistHandler:', error);

      throw error;

    }

  }

  async addSongToPlaylistHandler(request, h) {

    try {

      // Validasi format payload

      this.\_validator.validatePlaylistSongPayload(request.payload);

      const { playlistId } = request.params;

      const { id: credentialId } = request.auth.credentials;

      const { songId } = request.payload;

      // Verifikasi kepemilikan playlist

      await this.\_service.verifyPlaylistOwner(playlistId, credentialId);

      // Verifikasi bahwa lagu ada di database

      await this.\_playlistSongsService.verifySongExists(songId);

      // Tambahkan lagu ke playlist

      await this.\_playlistSongsService.addSongToPlaylist(playlistId, songId);

      return h.response({

        status: 'success',

        message: 'Lagu berhasil ditambahkan ke playlist',

      }).code(201);

    } catch (error) {

      console.error('Error in addSongToPlaylistHandler:', error);

      throw error;

    }

  }

  async getSongsFromPlaylistHandler(request, h) {

    try {

      const { playlistId } = request.params;

      const { id: credentialId } = request.auth.credentials;

      console.log('[DEBUG] playlistId:', playlistId);

      console.log('[DEBUG] credentialId:', credentialId);

      // Verifikasi kepemilikan atau akses playlist

      await this.\_service.verifyPlaylistOwner(playlistId, credentialId);

      console.log('[DEBUG] Playlist ownership verified');

      const songs = await this.\_playlistSongsService.getSongsFromPlaylist(playlistId);

      console.log('[DEBUG] Retrieved songs:', songs);

      return {

        status: 'success',

        data: {

          songs,

        },

      };

    } catch (error) {

      console.error('Error in getSongsFromPlaylistHandler:', error);

      throw error;

    }

  }

}

module.exports = PlaylistsHandler;

>routes.js

const routes = (handler) => [

  {

    method: 'POST',

    path: '/playlists',

    handler: handler.postPlaylistHandler,

    options: {

      auth: 'openmusic\_jwt',

    },

  },

  {

    method: 'GET',

    path: '/playlists',

    handler: handler.getPlaylistsHandler,

    options: {

      auth: 'openmusic\_jwt',

    },

  },

{

  method: 'GET',

  path: '/playlists/{playlistId}/songs',

  handler: handler.getSongsFromPlaylistHandler,

  options: {

    auth: 'openmusic\_jwt',

  },

},

  {

    method: 'DELETE',

    path: '/playlists/{id}',

    handler: handler.deletePlaylistHandler,

    options: {

      auth: 'openmusic\_jwt',

    },

  },

  {

    method: 'POST',

    path: '/playlists/{playlistId}/songs',

    handler: handler.addSongToPlaylistHandler,

    options: {

      auth: 'openmusic\_jwt',

    },

  },

];

module.exports = routes;

>validatorPlaylists.js

const Joi = require('joi');

const InvariantError = require('../../exceptions/InvariantError');

/\* ✅ Schema untuk membuat playlist \*/

const PlaylistPayloadSchema = Joi.object({

  name: Joi.string().required(),

});

/\* ✅ Schema untuk menambahkan lagu ke playlist \*/

const PlaylistSongPayloadSchema = Joi.object({

  songId: Joi.string().required(),

});

const PlaylistsValidator = {

  validatePlaylistPayload: (payload) => {

    const result = PlaylistPayloadSchema.validate(payload);

    if (result.error) {

      throw new InvariantError(result.error.message);

    }

  },

  validatePlaylistSongPayload: (payload) => {

    const result = PlaylistSongPayloadSchema.validate(payload);

    if (result.error) {

      throw new InvariantError(result.error.message);

    }

  },

};

module.exports = PlaylistsValidator;

>songs

>handlerSongs.js

const autoBind = require('auto-bind');

class SongsHandler {

  constructor(service, validator) {

    this.\_service = service;

    this.\_validator = validator;

    autoBind(this); // ⬅️ menggantikan semua manual .bind()

  }

  async postSongHandler(request, h) {

    try {

      this.\_validator.validateSongPayload(request.payload);

      const { title, year, genre, performer, duration, albumId } = request.payload;

      const songId = await this.\_service.addSong({ title, year, genre, performer, duration, albumId });

      const response = h.response({

        status: 'success',

        data: { songId },

      });

      response.code(201);

      return response;

    } catch (error) {

      console.error('Error in postSongHandler:', error);

      throw error;

    }

  }

  async getSongsHandler(request, h) {

    try {

      const { title, performer } = request.query;

      const songs = await this.\_service.getSongs({ title, performer });

      return {

        status: 'success',

        data: { songs },

      };

    } catch (error) {

      console.error('Error in getSongsHandler:', error);

      throw error;

    }

  }

  async getSongByIdHandler(request) {

    try {

      const { id } = request.params;

      const song = await this.\_service.getSongById(id);

      return {

        status: 'success',

        data: { song },

      };

    } catch (error) {

      console.error('Error in getSongByIdHandler:', error);

      throw error;

    }

  }

  async putSongByIdHandler(request) {

    try {

      this.\_validator.validateSongPayload(request.payload);

      const { id } = request.params;

      await this.\_service.editSongById(id, request.payload);

      return {

        status: 'success',

        message: 'Lagu berhasil diperbarui',

      };

    } catch (error) {

      console.error('Error in putSongByIdHandler:', error);

      throw error;

    }

  }

  async deleteSongByIdHandler(request) {

    try {

      const { id } = request.params;

      await this.\_service.deleteSongById(id);

      return {

        status: 'success',

        message: 'Lagu berhasil dihapus',

      };

    } catch (error) {

      console.error('Error in deleteSongByIdHandler:', error);

      throw error;

    }

  }

}

module.exports = SongsHandler;

>index.js

const SongsHandler = require('./handler');

const routes = require('./routes');

module.exports = {

  name: 'songs',

  version: '1.0.0',

  register: async (server, { songsService, songsValidator }) => {

    const songsHandler = new SongsHandler(songsService, songsValidator);

    server.route(routes(songsHandler));

  },

};

>routeSongs.js

const routes = (handler) => [

  {

    method: 'POST',

    path: '/songs',

    handler: handler.postSongHandler,

  },

  {

    method: 'GET',

    path: '/songs',

    handler: handler.getSongsHandler,

  },

  {

    method: 'GET',

    path: '/songs/{id}',

    handler: handler.getSongByIdHandler,

  },

  {

    method: 'PUT',

    path: '/songs/{id}',

    handler: handler.putSongByIdHandler,

  },

  {

    method: 'DELETE',

    path: '/songs/{id}',

    handler: handler.deleteSongByIdHandler,

  },

];

module.exports = routes;

>validatorSongs.js

const Joi = require('joi');

const InvariantError = require('../../exceptions/InvariantError');

const SongPayloadSchema = Joi.object({

  title: Joi.string().required(),

  year: Joi.number().required(),

  genre: Joi.string().required(),

  performer: Joi.string().required(),

  duration: Joi.number().optional(),

  albumId: Joi.string().optional(),

});

const SongsValidator = {

  validateSongPayload: (payload) => {

    const validationResult = SongPayloadSchema.validate(payload);

    if (validationResult.error) {

      throw new InvariantError(validationResult.error.message);

    }

  },

};

module.exports = SongsValidator;

>users

>handler.js

const autoBind = require('auto-bind');

class UsersHandler {

  constructor(service, validator) {

    this.\_service = service;

    this.\_validator = validator;

    autoBind(this);

  }

  async postUserHandler(request, h) {

    this.\_validator.validateUserPayload(request.payload);

    const { username, password, fullname } = request.payload;

    const userId = await this.\_service.addUser({ username, password, fullname });

    const response = h.response({

      status: 'success',

      data: {

        userId,

      },

    });

    response.code(201);

    return response;

  }

}

module.exports = UsersHandler;

>index.js

const UsersHandler = require('./handler');

const routes = require('./routes');

module.exports = {

  name: 'users',

  version: '1.0.0',

  register: async (server, { usersService, usersValidator }) => {

    const usersHandler = new UsersHandler(usersService, usersValidator);

    server.route(routes(usersHandler));

  },

};

>routes.js

const routes = (handler) => [

  {

    method: 'POST',

    path: '/users',

    handler: handler.postUserHandler,

  },

];

module.exports = routes;

>validatorUsers.js

const Joi = require('joi');

const InvariantError = require('../../exceptions/InvariantError');

const UserPayloadSchema = Joi.object({

  username: Joi.string().required(),

  password: Joi.string().required(),

  fullname: Joi.string().required(),

});

const UsersValidator = {

  validateUserPayload: (payload) => {

    const validationResult = UserPayloadSchema.validate(payload);

    if (validationResult.error) {

      throw new InvariantError(validationResult.error.message);

    }

  },

};

module.exports = UsersValidator;

>exceptions

>AuthenticationError.js

const ClientError = require('./ClientError');

class AuthenticationError extends ClientError {

  constructor(message) {

    super(message);

    this.name = 'AuthenticationError';

    this.statusCode = 401;

  }

}

module.exports = AuthenticationError;

>AuthorizationError.js

const ClientError = require('./ClientError');

class AuthorizationError extends ClientError {

  constructor(message) {

    super(message, 403);

    this.name = 'AuthorizationError';

  }

}

module.exports = AuthorizationError;

>ClientError.js

class ClientError extends Error {

  constructor(message, statusCode = 400) {

    super(message);

    this.statusCode = statusCode;

    this.name = 'ClientError';

  }

}

module.exports = ClientError;

>InvariantError.js

const ClientError = require('./ClientError');

class InvariantError extends ClientError {

  constructor(message) {

    super(message, 400);

    this.name = 'InvariantError';

    console.log('InvariantError inherits from ClientError?', this instanceof ClientError);

  }

}

module.exports = InvariantError;

>NotFoundError.js

const ClientError = require('./ClientError');

class NotFoundError extends ClientError {

  constructor(message) {

    super(message, 404);

    this.name = 'NotFoundError';

  }

}

module.exports = NotFoundError;

>models

>db.js

const { Pool } = require('pg');

const dotenv = require('dotenv');

dotenv.config();

const pool = new Pool({

  host: process.env.PGHOST,

  port: process.env.PGPORT,

  database: process.env.PGDATABASE,

  user: process.env.PGUSER,

  password: process.env.PGPASSWORD,

});

module.exports = pool;

>services

>albumsService.js

const { nanoid } = require('nanoid');

const db = require('../models/db');

const NotFoundError = require('../exceptions/NotFoundError');

class AlbumsService {

  async addAlbum({ name, year }) {

    const id = `album-${nanoid(16)}`;

    const query = {

      text: 'INSERT INTO albums (id, name, year) VALUES ($1, $2, $3) RETURNING id',

      values: [id, name, year],

    };

    const result = await db.query(query);

    return result.rows[0].id;

  }

  async getAlbums() {

    const result = await db.query('SELECT id, name FROM albums');

    return result.rows;

  }

  async getAlbumById(id) {

    const albumQuery = await db.query('SELECT \* FROM albums WHERE id = $1', [id]);

    if (!albumQuery.rowCount) {

      throw new NotFoundError('Album tidak ditemukan');

    }

    const songsQuery = await db.query(

      'SELECT id, title, performer FROM songs WHERE album\_id = $1',

      [id]

    );

    return {

      ...albumQuery.rows[0],

      songs: songsQuery.rows,

    };

  }

  async editAlbumById(id, { name, year }) {

    const query = {

      text: 'UPDATE albums SET name = $1, year = $2 WHERE id = $3 RETURNING id',

      values: [name, year, id],

    };

    const result = await db.query(query);

    if (!result.rowCount) {

      throw new NotFoundError('Gagal memperbarui album. Id tidak ditemukan');

    }

  }

  async deleteAlbumById(id) {

    const query = {

      text: 'DELETE FROM albums WHERE id = $1 RETURNING id',

      values: [id],

    };

    const result = await db.query(query);

    if (!result.rowCount) {

      throw new NotFoundError('Album gagal dihapus. Id tidak ditemukan');

    }

  }

}

module.exports = AlbumsService;

>authenticationsService.js

const InvariantError = require('../exceptions/InvariantError');

class AuthenticationsService {

  constructor() {

    this.\_refreshTokens = new Set(); // disimpan sementara di memori

  }

  addToken(token) {

    this.\_refreshTokens.add(token);

  }

  verifyRefreshToken(token) {

    if (!this.\_refreshTokens.has(token)) {

      throw new InvariantError('Refresh token tidak valid');

    }

  }

  deleteToken(token) {

    if (!this.\_refreshTokens.has(token)) {

      throw new InvariantError('Refresh token tidak ditemukan');

    }

    this.\_refreshTokens.delete(token);

  }

}

module.exports = AuthenticationsService;

>CollaborationsService.js

const { Pool } = require('pg');

const { nanoid } = require('nanoid');

const InvariantError = require('../exceptions/InvariantError');

const AuthorizationError = require('../exceptions/AuthorizationError');

class CollaborationsService {

  constructor() {

    this.\_pool = new Pool();

  }

  async addCollaboration(playlistId, userId) {

    const id = `collab-${nanoid(16)}`;

    const query = {

      text: 'INSERT INTO collaborations (id, playlist\_id, user\_id) VALUES ($1, $2, $3) RETURNING id',

      values: [id, playlistId, userId],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new InvariantError('Kolaborasi gagal ditambahkan');

    }

    return result.rows[0].id;

  }

  async deleteCollaboration(playlistId, userId) {

    const query = {

      text: 'DELETE FROM collaborations WHERE playlist\_id = $1 AND user\_id = $2 RETURNING id',

      values: [playlistId, userId],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new InvariantError('Kolaborasi gagal dihapus');

    }

  }

  async verifyCollaborator(playlistId, userId) {

    const query = {

      text: 'SELECT \* FROM collaborations WHERE playlist\_id = $1 AND user\_id = $2',

      values: [playlistId, userId],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new AuthorizationError('Anda tidak berhak mengakses resource ini');

    }

  }

}

module.exports = CollaborationsService;

>PlaylistSongsService.js

const { nanoid } = require('nanoid');

const { Pool } = require('pg');

const InvariantError = require('../exceptions/InvariantError');

const NotFoundError = require('../exceptions/NotFoundError'); // ✅ Ditambahkan

class PlaylistSongsService {

  constructor() {

    this.\_pool = new Pool();

  }

  async addSongToPlaylist(playlistId, songId) {

    const id = `playlistsong-${nanoid(16)}`;

    const query = {

      text: 'INSERT INTO playlist\_songs (id, playlist\_id, song\_id) VALUES($1, $2, $3) RETURNING id',

      values: [id, playlistId, songId],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new InvariantError('Gagal menambahkan lagu ke playlist');

    }

    return result.rows[0].id;

  }

  async getSongsFromPlaylist(playlistId) {

    const query = {

      text: `

        SELECT songs.id, songs.title, songs.performer

        FROM playlist\_songs

        JOIN songs ON songs.id = playlist\_songs.song\_id

        WHERE playlist\_songs.playlist\_id = $1

      `,

      values: [playlistId],

    };

    const result = await this.\_pool.query(query);

    return result.rows;

  }

  async deleteSongFromPlaylist(playlistId, songId) {

    const query = {

      text: 'DELETE FROM playlist\_songs WHERE playlist\_id = $1 AND song\_id = $2 RETURNING id',

      values: [playlistId, songId],

    };

    const result = await this.\_pool.query(query);

    if (!result.rowCount) {

      throw new InvariantError('Lagu gagal dihapus dari playlist. Id tidak ditemukan');

    }

  }

  async verifyPlaylistSong(playlistId, songId) {

    const query = {

      text: 'SELECT id FROM playlist\_songs WHERE playlist\_id = $1 AND song\_id = $2',

      values: [playlistId, songId],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new InvariantError('Lagu tidak ditemukan di dalam playlist');

    }

  }

  async verifySongExists(songId) {

    const result = await this.\_pool.query('SELECT id FROM songs WHERE id = $1', [songId]);

    if (!result.rowCount) {

      throw new NotFoundError('Lagu tidak ditemukan');

    }

  }

}

module.exports = PlaylistSongsService;

>PlaylistsService.js

const { nanoid } = require('nanoid');

const { Pool } = require('pg');

const InvariantError = require('../exceptions/InvariantError');

const NotFoundError = require('../exceptions/NotFoundError');

const AuthorizationError = require('../exceptions/AuthorizationError'); // ✅ Perbaikan di sini

class PlaylistsService {

  constructor() {

    this.\_pool = new Pool();

  }

  async addPlaylist({ name, owner }) {

    const id = `playlist-${nanoid(16)}`;

    const query = {

      text: 'INSERT INTO playlists (id, name, owner) VALUES ($1, $2, $3) RETURNING id',

      values: [id, name, owner],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new InvariantError('Gagal menambahkan playlist');

    }

    return result.rows[0].id;

  }

  async getPlaylists(owner) {

    const query = {

      text: `

        SELECT playlists.id, playlists.name, users.username

        FROM playlists

        JOIN users ON users.id = playlists.owner

        WHERE playlists.owner = $1

      `,

      values: [owner],

    };

    const result = await this.\_pool.query(query);

    return result.rows;

  }

  async deletePlaylist(id) {

    const query = {

      text: 'DELETE FROM playlists WHERE id = $1 RETURNING id',

      values: [id],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new NotFoundError('Playlist gagal dihapus. Id tidak ditemukan');

    }

  }

  async verifyPlaylistOwner(id, owner) {

    const query = {

      text: 'SELECT owner FROM playlists WHERE id = $1',

      values: [id],

    };

    const result = await this.\_pool.query(query);

    if (!result.rows.length) {

      throw new NotFoundError('Playlist tidak ditemukan');

    }

    const playlist = result.rows[0];

    if (playlist.owner !== owner) {

      throw new AuthorizationError('Anda tidak berhak mengakses resource ini'); // ✅ Diperbaiki di sini

    }

  }

}

module.exports = PlaylistsService;

>songsService.js

const { nanoid } = require('nanoid');

const db = require('../models/db');

const NotFoundError = require('../exceptions/NotFoundError');

class SongsService {

  constructor(albumsService) {

    this.\_albumsService = albumsService;

  }

  async addSong({ title, year, genre, performer, duration, albumId }) {

    // Validasi albumId jika disertakan

    if (albumId) {

      try {

        await this.\_albumsService.getAlbumById(albumId);

      } catch (error) {

        throw new NotFoundError('Album tidak ditemukan');

      }

    }

    const id = `song-${nanoid(16)}`;

    const query = {

      text: `INSERT INTO songs (id, title, year, genre, performer, duration, album\_id)

             VALUES ($1, $2, $3, $4, $5, $6, $7) RETURNING id`,

      values: [id, title, year, genre, performer, duration, albumId],

    };

    const result = await db.query(query);

    return result.rows[0].id;

  }

  async getSongs({title, performer}) {

    let baseQuery = 'SELECT id, title, performer FROM songs';

    const conditions = [];

    const values = [];

    if (title) {

      values.push(`%${title.toLowerCase()}%`);

    conditions.push(`LOWER(title) LIKE $${values.length}`);

    }

     if (performer) {

    values.push(`%${performer.toLowerCase()}%`);

    conditions.push(`LOWER(performer) LIKE $${values.length}`);

  }

   if (conditions.length > 0) {

    baseQuery += ' WHERE ' + conditions.join(' AND ');

  }

    if(!title && !performer){

      const result = await db.query('SELECT id, title, performer FROM songs');

    }

    const result = await db.query(baseQuery, values);

    return result.rows;

  }

  async getSongById(id) {

    const query = {

      text: 'SELECT \* FROM songs WHERE id = $1',

      values: [id],

    };

    const result = await db.query(query);

    if (!result.rowCount) throw new NotFoundError('Lagu tidak ditemukan');

    return result.rows[0];

  }

  async editSongById(id, { title, year, genre, performer, duration, albumId }) {

    const query = {

      text: `UPDATE songs SET title = $1, year = $2, genre = $3,

             performer = $4, duration = $5, album\_id = $6 WHERE id = $7 RETURNING id`,

      values: [title, year, genre, performer, duration, albumId, id],

    };

    const result = await db.query(query);

    if (!result.rowCount) throw new NotFoundError('Gagal memperbarui lagu. Id tidak ditemukan');

  }

  async deleteSongById(id) {

    const query = {

      text: 'DELETE FROM songs WHERE id = $1 RETURNING id',

      values: [id],

    };

    const result = await db.query(query);

    if (!result.rowCount) throw new NotFoundError('Lagu gagal dihapus. Id tidak ditemukan');

  }

}

module.exports = SongsService;

>usersService.js

const { nanoid } = require('nanoid');

const bcrypt = require('bcrypt');

const { Pool } = require('pg');

const InvariantError = require('../exceptions/InvariantError');

const NotFoundError = require('../exceptions/NotFoundError');

const AuthenticationError = require('../exceptions/AuthenticationError');

class UsersService {

  constructor() {

    this.\_pool = new Pool();

  }

  async addUser({ username, password, fullname }) {

    // Cek apakah username sudah ada di database

    const queryCheck = {

      text: 'SELECT username FROM users WHERE username = $1',

      values: [username],

    };

    const resultCheck = await this.\_pool.query(queryCheck);

    if (resultCheck.rowCount > 0) {

      throw new InvariantError('Gagal menambahkan user. Username sudah digunakan.');

    }

    const id = `user-${nanoid(16)}`;

    const hashedPassword = await bcrypt.hash(password, 10);

    const queryInsert = {

      text: 'INSERT INTO users (id, username, password, fullname) VALUES ($1, $2, $3, $4) RETURNING id',

      values: [id, username, hashedPassword, fullname],

    };

    const resultInsert = await this.\_pool.query(queryInsert);

    console.log('[DEBUG] Hasil insert user:', resultInsert.rows);

    if (!resultInsert.rows.length) {

      throw new InvariantError('Gagal menambahkan user');

    }

    return resultInsert.rows[0].id;

  }

  async verifyUserCredential(username, password) {

    const query = {

      text: 'SELECT id, password FROM users WHERE username = $1',

      values: [username],

    };

    const result = await this.\_pool.query(query);

    if (!result.rowCount) {

      throw new AuthenticationError('Kredensial yang Anda berikan salah');

    }

    const { id, password: hashedPassword } = result.rows[0];

    const isPasswordCorrect = await bcrypt.compare(password, hashedPassword);

    if (!isPasswordCorrect) {

      throw new AuthenticationError('Kredensial yang Anda berikan salah');

    }

    return id;

  }

}

module.exports = UsersService;

>tokenizes

>TokenManager.js

const Jwt = require('@hapi/jwt');

const jsonwebtoken = require('jsonwebtoken');

const InvariantError = require('../exceptions/InvariantError');

const TokenManager = {

  // 🔁 Generate Access Token (dengan log iat & exp untuk debugging)

  generateAccessToken: (payload) => {

    const token = Jwt.token.generate(payload, process.env.ACCESS\_TOKEN\_KEY, {

      ttlSec: parseInt(process.env.ACCESS\_TOKEN\_AGE, 10) || 86400, // fallback 24 jam

    });

    const decoded = Jwt.token.decode(token);

    console.log('Access Token:', token);

    console.log('iat:', decoded.decoded.payload.iat, new Date(decoded.decoded.payload.iat \* 1000));

    console.log('exp:', decoded.decoded.payload.exp, new Date(decoded.decoded.payload.exp \* 1000));

    return token;

  },

  // 🔁 Generate Refresh Token (pakai jsonwebtoken karena tidak expired)

  generateRefreshToken: (payload) =>

    jsonwebtoken.sign(payload, process.env.REFRESH\_TOKEN\_KEY),

  // 🔍 Verify Refresh Token

  verifyRefreshToken: (token) => {

    try {

      const decoded = jsonwebtoken.verify(token, process.env.REFRESH\_TOKEN\_KEY);

      return decoded;

    } catch (e) {

      throw new InvariantError('Refresh token tidak valid');

    }

  },

};

module.exports = TokenManager;

>utils

>nanoid.js

import { nanoid } from 'nanoid';

export default nanoid;

>app.js

const Hapi = require('@hapi/hapi');

const Jwt = require('@hapi/jwt');

/\* Services \*/

const AlbumsService = require('./services/albumsService');

const SongsService = require('./services/songsService');

const UsersService = require('./services/usersService');

const AuthenticationsService = require('./services/authenticationsService');

const PlaylistsService = require('./services/PlaylistsService');

const PlaylistSongsService = require('./services/PlaylistSongsService'); // ⬅️ ditambahkan

/\* Handlers & Validators \*/

const AlbumsHandler = require('./api/albums/handlerAlbums');

const SongsHandler = require('./api/songs/handlerSongs');

const AlbumsValidator = require('./api/albums/validatorAlbums');

const SongsValidator = require('./api/songs/validatorSongs');

const UsersValidator = require('./api/users/validatorUsers');

const AuthenticationsValidator = require('./api/authentications/validatorAuthentications');

const PlaylistsValidator = require('./api/playlists/validatorPlaylists');

/\* Token Manager \*/

const TokenManager = require('./tokenize/TokenManager');

/\* Plugins \*/

const users = require('./api/users');

const authentications = require('./api/authentications');

const playlists = require('./api/playlists');

/\* Routes non-plugin \*/

const albumsRoutes = require('./api/albums/routesAlbums');

const songsRoutes = require('./api/songs/routeSongs');

const ClientError = require('./exceptions/ClientError');

const init = async () => {

  /\* ---------- Inisialisasi service domain ---------- \*/

  const albumsService = new AlbumsService();

  const songsService = new SongsService(albumsService);

  const usersService = new UsersService();

  const authenticationsService = new AuthenticationsService();

  const playlistsService = new PlaylistsService();

  const playlistSongsService = new PlaylistSongsService(); // ⬅️ ditambahkan

  /\* ---------- Inisialisasi handler ---------- \*/

  const albumsHandler = new AlbumsHandler(albumsService, AlbumsValidator);

  const songsHandler = new SongsHandler(songsService, SongsValidator);

  /\* ---------- Membuat server ---------- \*/

  const server = Hapi.server({

    port: process.env.PORT || 9000,

    host: process.env.HOST || 'localhost',

    routes: {

      cors: { origin: ['\*'] },

      payload: { parse: true, output: 'data' },

    },

  });

  /\* ---------- Register plugin eksternal ---------- \*/

  await server.register([Jwt]);

  /\* ---------- Strategi autentikasi JWT ---------- \*/

  server.auth.strategy('openmusic\_jwt', 'jwt', {

    keys: process.env.ACCESS\_TOKEN\_KEY,

    verify: {

      aud: false,

      iss: false,

      sub: false,

      maxAgeSec: Number(process.env.ACCESS\_TOKEN\_AGE),

    },

    validate: (artifacts) => {

      const now = Math.floor(Date.now() / 1000);

      const { iat, exp } = artifacts.decoded.payload;

      console.log('[JWT DEBUG] Server now:', now, new Date(now \* 1000));

      console.log('[JWT DEBUG] Token iat:', iat, new Date(iat \* 1000));

      console.log('[JWT DEBUG] Token exp:', exp, new Date(exp \* 1000));

      return {

        isValid: true,

        credentials: {

          id: artifacts.decoded.payload.id,

        },

      };

    },

  });

  /\* ---------- Register route non-plugin ---------- \*/

  server.route(albumsRoutes(albumsHandler));

  server.route(songsRoutes(songsHandler));

  /\* ---------- Register plugin internal ---------- \*/

  await server.register([

    {

      plugin: users,

      options: {

        usersService,

        usersValidator: UsersValidator,

      },

    },

    {

      plugin: authentications,

      options: {

        authenticationsService,

        usersService,

        tokenManager: TokenManager,

        authenticationsValidator: AuthenticationsValidator,

      },

    },

    {

      plugin: playlists,

      options: {

        playlistsService,

        playlistsValidator: PlaylistsValidator,

        playlistSongsService, // ⬅️ integrasi ke plugin playlists

      },

    },

  ]);

  /\* ---------- Middleware error handling diperbarui ---------- \*/

  server.ext('onPreResponse', (request, h) => {

    const { response } = request;

    if (response instanceof Error) {

      console.error('[onPreResponse] ERROR TYPE:', response.name);

      console.error('[onPreResponse] MESSAGE:', response.message);

      console.error('[onPreResponse] STATUS CODE:', response.statusCode);

    }

    // ✅ Tangani token rusak (invalid)

    if (response.isBoom && response.message === 'Invalid token signature') {

      const newResponse = h.response({

        status: 'fail',

        message: 'Token tidak valid',

      });

      newResponse.code(401);

      return newResponse;

    }

    // ✅ Tangani token kedaluwarsa (expired)

    if (response.isBoom && response.message === 'Token expired') {

      const newResponse = h.response({

        status: 'fail',

        message: 'Token telah kedaluwarsa',

      });

      newResponse.code(401);

      return newResponse;

    }

    // ✅ Tangani token kosong (missing)

    if (response.isBoom && response.message === 'Missing authentication') {

      const newResponse = h.response({

        status: 'fail',

        message: 'Autentikasi diperlukan',

      });

      newResponse.code(401);

      return newResponse;

    }

    // ✅ Tangani format header Authorization yang salah

    if (response.isBoom && response.message === 'Bad HTTP authentication header format') {

      const newResponse = h.response({

        status: 'fail',

        message: 'Format header autentikasi salah',

      });

      newResponse.code(400);

      return newResponse;

    }

    if (response.name === 'InvariantError') {

      const newResponse = h.response({

        status: 'fail',

        message: response.message,

      });

      newResponse.code(400);

      return newResponse;

    }

    if (response instanceof ClientError) {

      const newResponse = h.response({

        status: 'fail',

        message: response.message,

      });

      newResponse.code(response.statusCode);

      return newResponse;

    }

    if (response.isBoom) {

      const newResponse = h.response({

        status: 'error',

        message: 'Maaf, terjadi kegagalan pada server kami.',

      });

      newResponse.code(500);

      return newResponse;

    }

    return h.continue;

  });

  return server;

};

module.exports = init;

>server.js

const dotenv = require('dotenv');

dotenv.config();

const init = require('./app');

init()

  .then((server) => {

    server.start();

    console.log(`Server berjalan pada ${server.info.uri}`);

  })

  .catch((err) => {

    console.error('Gagal memulai server:', err);

  });

.env

HOST=localhost

PORT=5000

# JWT Secrets & Config

ACCESS\_TOKEN\_KEY=tTsw+uAhVHR7V3SGsVWJYrrpM1LFLUnvMdZyAie8w+o=

REFRESH\_TOKEN\_KEY=K8vM7+49LrY0WnNtU8bNL0FxTGFYHmkIkE1PbMZl4no=

ACCESS\_TOKEN\_AGE=7200

# Database

PGUSER=postgres

PGPASSWORD=;P0lo9ki8

PGHOST=localhost

PGDATABASE=openmusic

PGPORT=5432

.gitignore

{} database.json

{

  "dev": {

    "driver": "pg",

    "host": "localhost",

    "port": 5432,

    "database": "openmusic",

    "user": "postgres",

    "password": ";P0lo9ki8"

  }

}

{} package-lock.json

{

"name": "openmusic-api",

"version": "1.0.0",

"lockfileVersion": 3,

"requires": true,

"packages": {

"": {

"name": "openmusic-api",

"version": "1.0.0",

"license": "ISC",

"dependencies": {

"@hapi/hapi": "^21.4.0",

"@hapi/jwt": "^3.2.0",

"auto-bind": "^4.0.0",

"bcrypt": "^6.0.0",

"dotenv": "^16.5.0",

"joi": "^17.13.3",

"jsonwebtoken": "^9.0.2",

"nanoid": "^3.3.11",

"pg": "^8.16.0"

},

"devDependencies": {

"node-pg-migrate": "^8.0.1",

"nodemon": "^3.1.10"

}

},

"node\_modules/@hapi/accept": {

"version": "6.0.3",

"resolved": "https://registry.npmjs.org/@hapi/accept/-/accept-6.0.3.tgz",

"integrity": "sha512-p72f9k56EuF0n3MwlBNThyVE5PXX40g+aQh+C/xbKrfzahM2Oispv3AXmOIU51t3j77zay1qrX7IIziZXspMlw==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/accept/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/ammo": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/@hapi/ammo/-/ammo-6.0.1.tgz",

"integrity": "sha512-pmL+nPod4g58kXrMcsGLp05O2jF4P2Q3GiL8qYV7nKYEh3cGf+rV4P5Jyi2Uq0agGhVU63GtaSAfBEZOlrJn9w==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/ammo/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/b64": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/@hapi/b64/-/b64-6.0.1.tgz",

"integrity": "sha512-ZvjX4JQReUmBheeCq+S9YavcnMMHWqx3S0jHNXWIM1kQDxB9cyfSycpVvjfrKcIS8Mh5N3hmu/YKo4Iag9g2Kw==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/b64/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/boom": {

"version": "10.0.1",

"resolved": "https://registry.npmjs.org/@hapi/boom/-/boom-10.0.1.tgz",

"integrity": "sha512-ERcCZaEjdH3OgSJlyjVk8pHIFeus91CjKP3v+MpgBNp5IvGzP2l/bRiD78nqYcKPaZdbKkK5vDBVPd2ohHBlsA==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/boom/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/bounce": {

"version": "3.0.2",

"resolved": "https://registry.npmjs.org/@hapi/bounce/-/bounce-3.0.2.tgz",

"integrity": "sha512-d0XmlTi3H9HFDHhQLjg4F4auL1EY3Wqj7j7/hGDhFFe6xAbnm3qiGrXeT93zZnPH8gH+SKAFYiRzu26xkXcH3g==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/bounce/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/bourne": {

"version": "3.0.0",

"resolved": "https://registry.npmjs.org/@hapi/bourne/-/bourne-3.0.0.tgz",

"integrity": "sha512-Waj1cwPXJDucOib4a3bAISsKJVb15MKi9IvmTI/7ssVEm6sywXGjVJDhl6/umt1pK1ZS7PacXU3A1PmFKHEZ2w==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/call": {

"version": "9.0.1",

"resolved": "https://registry.npmjs.org/@hapi/call/-/call-9.0.1.tgz",

"integrity": "sha512-uPojQRqEL1GRZR4xXPqcLMujQGaEpyVPRyBlD8Pp5rqgIwLhtveF9PkixiKru2THXvuN8mUrLeet5fqxKAAMGg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/call/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/catbox": {

"version": "12.1.1",

"resolved": "https://registry.npmjs.org/@hapi/catbox/-/catbox-12.1.1.tgz",

"integrity": "sha512-hDqYB1J+R0HtZg4iPH3LEnldoaBsar6bYp0EonBmNQ9t5CO+1CqgCul2ZtFveW1ReA5SQuze9GPSU7/aecERhw==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2",

"@hapi/podium": "^5.0.0",

"@hapi/validate": "^2.0.1"

}

},

"node\_modules/@hapi/catbox-memory": {

"version": "6.0.2",

"resolved": "https://registry.npmjs.org/@hapi/catbox-memory/-/catbox-memory-6.0.2.tgz",

"integrity": "sha512-H1l4ugoFW/ZRkqeFrIo8p1rWN0PA4MDTfu4JmcoNDvnY975o29mqoZblqFTotxNHlEkMPpIiIBJTV+Mbi+aF0g==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/catbox-memory/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/catbox-object": {

"version": "3.0.1",

"resolved": "https://registry.npmjs.org/@hapi/catbox-object/-/catbox-object-3.0.1.tgz",

"integrity": "sha512-3w6E2DXtjWbmLYi4WcFUOor5jgrXN4PWhDrMrXKP/cEsFSfVSRJ0FhY2PXrhrUHwcllfKezYafWU3tQ5+8RO1w==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/catbox-object/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/catbox/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/content": {

"version": "6.0.0",

"resolved": "https://registry.npmjs.org/@hapi/content/-/content-6.0.0.tgz",

"integrity": "sha512-CEhs7j+H0iQffKfe5Htdak5LBOz/Qc8TRh51cF+BFv0qnuph3Em4pjGVzJMkI2gfTDdlJKWJISGWS1rK34POGA==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.0"

}

},

"node\_modules/@hapi/cryptiles": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/@hapi/cryptiles/-/cryptiles-6.0.1.tgz",

"integrity": "sha512-9GM9ECEHfR8lk5ASOKG4+4ZsEzFqLfhiryIJ2ISePVB92OHLp/yne4m+zn7z9dgvM98TLpiFebjDFQ0UHcqxXQ==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1"

},

"engines": {

"node": ">=14.0.0"

}

},

"node\_modules/@hapi/file": {

"version": "3.0.0",

"resolved": "https://registry.npmjs.org/@hapi/file/-/file-3.0.0.tgz",

"integrity": "sha512-w+lKW+yRrLhJu620jT3y+5g2mHqnKfepreykvdOcl9/6up8GrQQn+l3FRTsjHTKbkbfQFkuksHpdv2EcpKcJ4Q==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/hapi": {

"version": "21.4.0",

"resolved": "https://registry.npmjs.org/@hapi/hapi/-/hapi-21.4.0.tgz",

"integrity": "sha512-kqiRWbYYLSSt2rYbxyNj8svPsXP715p4W/K3OXpXeiiVLNSdBX4f+zfmC+dY6eyb6rqTqTAbx6x8b5HpJTkviQ==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/accept": "^6.0.3",

"@hapi/ammo": "^6.0.1",

"@hapi/boom": "^10.0.1",

"@hapi/bounce": "^3.0.2",

"@hapi/call": "^9.0.1",

"@hapi/catbox": "^12.1.1",

"@hapi/catbox-memory": "^6.0.2",

"@hapi/heavy": "^8.0.1",

"@hapi/hoek": "^11.0.6",

"@hapi/mimos": "^7.0.1",

"@hapi/podium": "^5.0.1",

"@hapi/shot": "^6.0.1",

"@hapi/somever": "^4.1.1",

"@hapi/statehood": "^8.2.0",

"@hapi/subtext": "^8.1.0",

"@hapi/teamwork": "^6.0.0",

"@hapi/topo": "^6.0.2",

"@hapi/validate": "^2.0.1"

},

"engines": {

"node": ">=14.15.0"

}

},

"node\_modules/@hapi/hapi/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/hapi/node\_modules/@hapi/topo": {

"version": "6.0.2",

"resolved": "https://registry.npmjs.org/@hapi/topo/-/topo-6.0.2.tgz",

"integrity": "sha512-KR3rD5inZbGMrHmgPxsJ9dbi6zEK+C3ZwUwTa+eMwWLz7oijWUTWD2pMSNNYJAU6Qq+65NkxXjqHr/7LM2Xkqg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/heavy": {

"version": "8.0.1",

"resolved": "https://registry.npmjs.org/@hapi/heavy/-/heavy-8.0.1.tgz",

"integrity": "sha512-gBD/NANosNCOp6RsYTsjo2vhr5eYA3BEuogk6cxY0QdhllkkTaJFYtTXv46xd6qhBVMbMMqcSdtqey+UQU3//w==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/hoek": "^11.0.2",

"@hapi/validate": "^2.0.1"

}

},

"node\_modules/@hapi/heavy/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/hoek": {

"version": "9.3.0",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-9.3.0.tgz",

"integrity": "sha512-/c6rf4UJlmHlC9b5BaNvzAcFv7HZ2QHaV0D4/HNlBdvFnvQq8RI4kYdhyPCl7Xj+oWvTWQ8ujhqS53LIgAe6KQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/iron": {

"version": "7.0.1",

"resolved": "https://registry.npmjs.org/@hapi/iron/-/iron-7.0.1.tgz",

"integrity": "sha512-tEZnrOujKpS6jLKliyWBl3A9PaE+ppuL/+gkbyPPDb/l2KSKQyH4lhMkVb+sBhwN+qaxxlig01JRqB8dk/mPxQ==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/b64": "^6.0.1",

"@hapi/boom": "^10.0.1",

"@hapi/bourne": "^3.0.0",

"@hapi/cryptiles": "^6.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/iron/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/jwt": {

"version": "3.2.0",

"resolved": "https://registry.npmjs.org/@hapi/jwt/-/jwt-3.2.0.tgz",

"integrity": "sha512-2EU5zdr0petG63J2RHiagGByA1Qr6qzxMmrJJ3/0cm78be1Lq4vi038YgKJvbBSxSboIp5SWGZdYB6+CJlqEoQ==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/b64": "^6.0.0",

"@hapi/boom": "^10.0.0",

"@hapi/bounce": "^3.0.0",

"@hapi/bourne": "^3.0.0",

"@hapi/catbox-object": "^3.0.0",

"@hapi/cryptiles": "^6.0.0",

"@hapi/hoek": "^10.0.0",

"@hapi/wreck": "^18.0.0",

"ecdsa-sig-formatter": "^1.0.0",

"joi": "^17.2.1"

}

},

"node\_modules/@hapi/jwt/node\_modules/@hapi/hoek": {

"version": "10.0.1",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-10.0.1.tgz",

"integrity": "sha512-CvlW7jmOhWzuqOqiJQ3rQVLMcREh0eel4IBnxDx2FAcK8g7qoJRQK4L1CPBASoCY6y8e6zuCy3f2g+HWdkzcMw==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/mimos": {

"version": "7.0.1",

"resolved": "https://registry.npmjs.org/@hapi/mimos/-/mimos-7.0.1.tgz",

"integrity": "sha512-b79V+BrG0gJ9zcRx1VGcCI6r6GEzzZUgiGEJVoq5gwzuB2Ig9Cax8dUuBauQCFKvl2YWSWyOc8mZ8HDaJOtkew==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2",

"mime-db": "^1.52.0"

}

},

"node\_modules/@hapi/mimos/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/nigel": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/@hapi/nigel/-/nigel-5.0.1.tgz",

"integrity": "sha512-uv3dtYuB4IsNaha+tigWmN8mQw/O9Qzl5U26Gm4ZcJVtDdB1AVJOwX3X5wOX+A07qzpEZnOMBAm8jjSqGsU6Nw==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2",

"@hapi/vise": "^5.0.1"

},

"engines": {

"node": ">=14.0.0"

}

},

"node\_modules/@hapi/nigel/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/pez": {

"version": "6.1.0",

"resolved": "https://registry.npmjs.org/@hapi/pez/-/pez-6.1.0.tgz",

"integrity": "sha512-+FE3sFPYuXCpuVeHQ/Qag1b45clR2o54QoonE/gKHv9gukxQ8oJJZPR7o3/ydDTK6racnCJXxOyT1T93FCJMIg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/b64": "^6.0.1",

"@hapi/boom": "^10.0.1",

"@hapi/content": "^6.0.0",

"@hapi/hoek": "^11.0.2",

"@hapi/nigel": "^5.0.1"

}

},

"node\_modules/@hapi/pez/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/podium": {

"version": "5.0.2",

"resolved": "https://registry.npmjs.org/@hapi/podium/-/podium-5.0.2.tgz",

"integrity": "sha512-T7gf2JYHQQfEfewTQFbsaXoZxSvuXO/QBIGljucUQ/lmPnTTNAepoIKOakWNVWvo2fMEDjycu77r8k6dhreqHA==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2",

"@hapi/teamwork": "^6.0.0",

"@hapi/validate": "^2.0.1"

}

},

"node\_modules/@hapi/podium/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/shot": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/@hapi/shot/-/shot-6.0.1.tgz",

"integrity": "sha512-s5ynMKZXYoDd3dqPw5YTvOR/vjHvMTxc388+0qL0jZZP1+uwXuUD32o9DuuuLsmTlyXCWi02BJl1pBpwRuUrNA==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2",

"@hapi/validate": "^2.0.1"

}

},

"node\_modules/@hapi/shot/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/somever": {

"version": "4.1.1",

"resolved": "https://registry.npmjs.org/@hapi/somever/-/somever-4.1.1.tgz",

"integrity": "sha512-lt3QQiDDOVRatS0ionFDNrDIv4eXz58IibQaZQDOg4DqqdNme8oa0iPWcE0+hkq/KTeBCPtEOjDOBKBKwDumVg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/bounce": "^3.0.1",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/somever/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/statehood": {

"version": "8.2.0",

"resolved": "https://registry.npmjs.org/@hapi/statehood/-/statehood-8.2.0.tgz",

"integrity": "sha512-63JlCVIrsmuunWsyc3OeuFO+gH6v56swLCl7OM1w09l/exQKPUxSUDF2Slkuw8k91nIzr0A2/aPvjLOWf9ksrg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/bounce": "^3.0.1",

"@hapi/bourne": "^3.0.0",

"@hapi/cryptiles": "^6.0.1",

"@hapi/hoek": "^11.0.2",

"@hapi/iron": "^7.0.1",

"@hapi/validate": "^2.0.1"

}

},

"node\_modules/@hapi/statehood/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/subtext": {

"version": "8.1.0",

"resolved": "https://registry.npmjs.org/@hapi/subtext/-/subtext-8.1.0.tgz",

"integrity": "sha512-PyaN4oSMtqPjjVxLny1k0iYg4+fwGusIhaom9B2StinBclHs7v46mIW706Y+Wo21lcgulGyXbQrmT/w4dus6ww==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/bourne": "^3.0.0",

"@hapi/content": "^6.0.0",

"@hapi/file": "^3.0.0",

"@hapi/hoek": "^11.0.2",

"@hapi/pez": "^6.1.0",

"@hapi/wreck": "^18.0.1"

}

},

"node\_modules/@hapi/subtext/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/teamwork": {

"version": "6.0.0",

"resolved": "https://registry.npmjs.org/@hapi/teamwork/-/teamwork-6.0.0.tgz",

"integrity": "sha512-05HumSy3LWfXpmJ9cr6HzwhAavrHkJ1ZRCmNE2qJMihdM5YcWreWPfyN0yKT2ZjCM92au3ZkuodjBxOibxM67A==",

"license": "BSD-3-Clause",

"engines": {

"node": ">=14.0.0"

}

},

"node\_modules/@hapi/topo": {

"version": "5.1.0",

"resolved": "https://registry.npmjs.org/@hapi/topo/-/topo-5.1.0.tgz",

"integrity": "sha512-foQZKJig7Ob0BMAYBfcJk8d77QtOe7Wo4ox7ff1lQYoNNAb6jwcY1ncdoy2e9wQZzvNy7ODZCYJkK8kzmcAnAg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^9.0.0"

}

},

"node\_modules/@hapi/validate": {

"version": "2.0.1",

"resolved": "https://registry.npmjs.org/@hapi/validate/-/validate-2.0.1.tgz",

"integrity": "sha512-NZmXRnrSLK8MQ9y/CMqE9WSspgB9xA41/LlYR0k967aSZebWr4yNrpxIbov12ICwKy4APSlWXZga9jN5p6puPA==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2",

"@hapi/topo": "^6.0.1"

}

},

"node\_modules/@hapi/validate/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/validate/node\_modules/@hapi/topo": {

"version": "6.0.2",

"resolved": "https://registry.npmjs.org/@hapi/topo/-/topo-6.0.2.tgz",

"integrity": "sha512-KR3rD5inZbGMrHmgPxsJ9dbi6zEK+C3ZwUwTa+eMwWLz7oijWUTWD2pMSNNYJAU6Qq+65NkxXjqHr/7LM2Xkqg==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/vise": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/@hapi/vise/-/vise-5.0.1.tgz",

"integrity": "sha512-XZYWzzRtINQLedPYlIkSkUr7m5Ddwlu99V9elh8CSygXstfv3UnWIXT0QD+wmR0VAG34d2Vx3olqcEhRRoTu9A==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/vise/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@hapi/wreck": {

"version": "18.1.0",

"resolved": "https://registry.npmjs.org/@hapi/wreck/-/wreck-18.1.0.tgz",

"integrity": "sha512-0z6ZRCmFEfV/MQqkQomJ7sl/hyxvcZM7LtuVqN3vdAO4vM9eBbowl0kaqQj9EJJQab+3Uuh1GxbGIBFy4NfJ4w==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/boom": "^10.0.1",

"@hapi/bourne": "^3.0.0",

"@hapi/hoek": "^11.0.2"

}

},

"node\_modules/@hapi/wreck/node\_modules/@hapi/hoek": {

"version": "11.0.7",

"resolved": "https://registry.npmjs.org/@hapi/hoek/-/hoek-11.0.7.tgz",

"integrity": "sha512-HV5undWkKzcB4RZUusqOpcgxOaq6VOAH7zhhIr2g3G8NF/MlFO75SjOr2NfuSx0Mh40+1FqCkagKLJRykUWoFQ==",

"license": "BSD-3-Clause"

},

"node\_modules/@isaacs/cliui": {

"version": "8.0.2",

"resolved": "https://registry.npmjs.org/@isaacs/cliui/-/cliui-8.0.2.tgz",

"integrity": "sha512-O8jcjabXaleOG9DQ0+ARXWZBTfnP4WNAqzuiJK7ll44AmxGKv/J2M4TPjxjY3znBCfvBXFzucm1twdyFybFqEA==",

"dev": true,

"license": "ISC",

"dependencies": {

"string-width": "^5.1.2",

"string-width-cjs": "npm:string-width@^4.2.0",

"strip-ansi": "^7.0.1",

"strip-ansi-cjs": "npm:strip-ansi@^6.0.1",

"wrap-ansi": "^8.1.0",

"wrap-ansi-cjs": "npm:wrap-ansi@^7.0.0"

},

"engines": {

"node": ">=12"

}

},

"node\_modules/@sideway/address": {

"version": "4.1.5",

"resolved": "https://registry.npmjs.org/@sideway/address/-/address-4.1.5.tgz",

"integrity": "sha512-IqO/DUQHUkPeixNQ8n0JA6102hT9CmaljNTPmQ1u8MEhBo/R4Q8eKLN/vGZxuebwOroDB4cbpjheD4+/sKFK4Q==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^9.0.0"

}

},

"node\_modules/@sideway/formula": {

"version": "3.0.1",

"resolved": "https://registry.npmjs.org/@sideway/formula/-/formula-3.0.1.tgz",

"integrity": "sha512-/poHZJJVjx3L+zVD6g9KgHfYnb443oi7wLu/XKojDviHy6HOEOA6z1Trk5aR1dGcmPenJEgb2sK2I80LeS3MIg==",

"license": "BSD-3-Clause"

},

"node\_modules/@sideway/pinpoint": {

"version": "2.0.0",

"resolved": "https://registry.npmjs.org/@sideway/pinpoint/-/pinpoint-2.0.0.tgz",

"integrity": "sha512-RNiOoTPkptFtSVzQevY/yWtZwf/RxyVnPy/OcA9HBM3MlGDnBEYL5B41H0MTn0Uec8Hi+2qUtTfG2WWZBmMejQ==",

"license": "BSD-3-Clause"

},

"node\_modules/ansi-regex": {

"version": "6.1.0",

"resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-6.1.0.tgz",

"integrity": "sha512-7HSX4QQb4CspciLpVFwyRe79O3xsIZDDLER21kERQ71oaPodF8jL725AgJMFAYbooIqolJoRLuM81SpeUkpkvA==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=12"

},

"funding": {

"url": "https://github.com/chalk/ansi-regex?sponsor=1"

}

},

"node\_modules/ansi-styles": {

"version": "6.2.1",

"resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-styles-6.2.1.tgz",

"integrity": "sha512-bN798gFfQX+viw3R7yrGWRqnrN2oRkEkUjjl4JNn4E8GxxbjtG3FbrEIIY3l8/hrwUwIeCZvi4QuOTP4MErVug==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=12"

},

"funding": {

"url": "https://github.com/chalk/ansi-styles?sponsor=1"

}

},

"node\_modules/anymatch": {

"version": "3.1.3",

"resolved": "https://registry.npmjs.org/anymatch/-/anymatch-3.1.3.tgz",

"integrity": "sha512-KMReFUr0B4t+D+OBkjR3KYqvocp2XaSzO55UcB6mgQMd3KbcE+mWTyvVV7D/zsdEbNnV6acZUutkiHQXvTr1Rw==",

"dev": true,

"license": "ISC",

"dependencies": {

"normalize-path": "^3.0.0",

"picomatch": "^2.0.4"

},

"engines": {

"node": ">= 8"

}

},

"node\_modules/auto-bind": {

"version": "4.0.0",

"resolved": "https://registry.npmjs.org/auto-bind/-/auto-bind-4.0.0.tgz",

"integrity": "sha512-Hdw8qdNiqdJ8LqT0iK0sVzkFbzg6fhnQqqfWhBDxcHZvU75+B+ayzTy8x+k5Ix0Y92XOhOUlx74ps+bA6BeYMQ==",

"license": "MIT",

"engines": {

"node": ">=8"

},

"funding": {

"url": "https://github.com/sponsors/sindresorhus"

}

},

"node\_modules/balanced-match": {

"version": "1.0.2",

"resolved": "https://registry.npmjs.org/balanced-match/-/balanced-match-1.0.2.tgz",

"integrity": "sha512-3oSeUO0TMV67hN1AmbXsK4yaqU7tjiHlbxRDZOpH0KW9+CeX4bRAaX0Anxt0tx2MrpRpWwQaPwIlISEJhYU5Pw==",

"dev": true,

"license": "MIT"

},

"node\_modules/bcrypt": {

"version": "6.0.0",

"resolved": "https://registry.npmjs.org/bcrypt/-/bcrypt-6.0.0.tgz",

"integrity": "sha512-cU8v/EGSrnH+HnxV2z0J7/blxH8gq7Xh2JFT6Aroax7UohdmiJJlxApMxtKfuI7z68NvvVcmR78k2LbT6efhRg==",

"hasInstallScript": true,

"license": "MIT",

"dependencies": {

"node-addon-api": "^8.3.0",

"node-gyp-build": "^4.8.4"

},

"engines": {

"node": ">= 18"

}

},

"node\_modules/binary-extensions": {

"version": "2.3.0",

"resolved": "https://registry.npmjs.org/binary-extensions/-/binary-extensions-2.3.0.tgz",

"integrity": "sha512-Ceh+7ox5qe7LJuLHoY0feh3pHuUDHAcRUeyL2VYghZwfpkNIy/+8Ocg0a3UuSoYzavmylwuLWQOf3hl0jjMMIw==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

},

"funding": {

"url": "https://github.com/sponsors/sindresorhus"

}

},

"node\_modules/brace-expansion": {

"version": "2.0.1",

"resolved": "https://registry.npmjs.org/brace-expansion/-/brace-expansion-2.0.1.tgz",

"integrity": "sha512-XnAIvQ8eM+kC6aULx6wuQiwVsnzsi9d3WxzV3FpWTGA19F621kwdbsAcFKXgKUHZWsy+mY6iL1sHTxWEFCytDA==",

"dev": true,

"license": "MIT",

"dependencies": {

"balanced-match": "^1.0.0"

}

},

"node\_modules/braces": {

"version": "3.0.3",

"resolved": "https://registry.npmjs.org/braces/-/braces-3.0.3.tgz",

"integrity": "sha512-yQbXgO/OSZVD2IsiLlro+7Hf6Q18EJrKSEsdoMzKePKXct3gvD8oLcOQdIzGupr5Fj+EDe8gO/lxc1BzfMpxvA==",

"dev": true,

"license": "MIT",

"dependencies": {

"fill-range": "^7.1.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/buffer-equal-constant-time": {

"version": "1.0.1",

"resolved": "https://registry.npmjs.org/buffer-equal-constant-time/-/buffer-equal-constant-time-1.0.1.tgz",

"integrity": "sha512-zRpUiDwd/xk6ADqPMATG8vc9VPrkck7T07OIx0gnjmJAnHnTVXNQG3vfvWNuiZIkwu9KrKdA1iJKfsfTVxE6NA==",

"license": "BSD-3-Clause"

},

"node\_modules/chokidar": {

"version": "3.6.0",

"resolved": "https://registry.npmjs.org/chokidar/-/chokidar-3.6.0.tgz",

"integrity": "sha512-7VT13fmjotKpGipCW9JEQAusEPE+Ei8nl6/g4FBAmIm0GOOLMua9NDDo/DWp0ZAxCr3cPq5ZpBqmPAQgDda2Pw==",

"dev": true,

"license": "MIT",

"dependencies": {

"anymatch": "~3.1.2",

"braces": "~3.0.2",

"glob-parent": "~5.1.2",

"is-binary-path": "~2.1.0",

"is-glob": "~4.0.1",

"normalize-path": "~3.0.0",

"readdirp": "~3.6.0"

},

"engines": {

"node": ">= 8.10.0"

},

"funding": {

"url": "https://paulmillr.com/funding/"

},

"optionalDependencies": {

"fsevents": "~2.3.2"

}

},

"node\_modules/cliui": {

"version": "8.0.1",

"resolved": "https://registry.npmjs.org/cliui/-/cliui-8.0.1.tgz",

"integrity": "sha512-BSeNnyus75C4//NQ9gQt1/csTXyo/8Sb+afLAkzAptFuMsod9HFokGNudZpi/oQV73hnVK+sR+5PVRMd+Dr7YQ==",

"dev": true,

"license": "ISC",

"dependencies": {

"string-width": "^4.2.0",

"strip-ansi": "^6.0.1",

"wrap-ansi": "^7.0.0"

},

"engines": {

"node": ">=12"

}

},

"node\_modules/cliui/node\_modules/ansi-regex": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.1.tgz",

"integrity": "sha512-quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMWv37qOOb9pdJlMUEKFQ==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/cliui/node\_modules/ansi-styles": {

"version": "4.3.0",

"resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-styles-4.3.0.tgz",

"integrity": "sha512-zbB9rCJAT1rbjiVDb2hqKFHNYLxgtk8NURxZ3IZwD3F6NtxbXZQCnnSi1Lkx+IDohdPlFp222wVALIheZJQSEg==",

"dev": true,

"license": "MIT",

"dependencies": {

"color-convert": "^2.0.1"

},

"engines": {

"node": ">=8"

},

"funding": {

"url": "https://github.com/chalk/ansi-styles?sponsor=1"

}

},

"node\_modules/cliui/node\_modules/emoji-regex": {

"version": "8.0.0",

"resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-regex-8.0.0.tgz",

"integrity": "sha512-MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8Eu3TdM14SawnVUmGE6A==",

"dev": true,

"license": "MIT"

},

"node\_modules/cliui/node\_modules/string-width": {

"version": "4.2.3",

"resolved": "https://registry.npmjs.org/string-width/-/string-width-4.2.3.tgz",

"integrity": "sha512-wKyQRQpjJ0sIp62ErSZdGsjMJWsap5oRNihHhu6G7JVO/9jIB6UyevL+tXuOqrng8j/cxKTWyWUwvSTriiZz/g==",

"dev": true,

"license": "MIT",

"dependencies": {

"emoji-regex": "^8.0.0",

"is-fullwidth-code-point": "^3.0.0",

"strip-ansi": "^6.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/cliui/node\_modules/strip-ansi": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-6.0.1.tgz",

"integrity": "sha512-Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/0ooI7KrPuUSztUdU5A==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-regex": "^5.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/cliui/node\_modules/wrap-ansi": {

"version": "7.0.0",

"resolved": "https://registry.npmjs.org/wrap-ansi/-/wrap-ansi-7.0.0.tgz",

"integrity": "sha512-YVGIj2kamLSTxw6NsZjoBxfSwsn0ycdesmc4p+Q21c5zPuZ1pl+NfxVdxPtdHvmNVOQ6XSYG4AUtyt/Fi7D16Q==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-styles": "^4.0.0",

"string-width": "^4.1.0",

"strip-ansi": "^6.0.0"

},

"engines": {

"node": ">=10"

},

"funding": {

"url": "https://github.com/chalk/wrap-ansi?sponsor=1"

}

},

"node\_modules/color-convert": {

"version": "2.0.1",

"resolved": "https://registry.npmjs.org/color-convert/-/color-convert-2.0.1.tgz",

"integrity": "sha512-RRECPsj7iu/xb5oKYcsFHSppFNnsj/52OVTRKb4zP5onXwVF3zVmmToNcOfGC+CRDpfK/U584fMg38ZHCaElKQ==",

"dev": true,

"license": "MIT",

"dependencies": {

"color-name": "~1.1.4"

},

"engines": {

"node": ">=7.0.0"

}

},

"node\_modules/color-name": {

"version": "1.1.4",

"resolved": "https://registry.npmjs.org/color-name/-/color-name-1.1.4.tgz",

"integrity": "sha512-dOy+3AuW3a2wNbZHIuMZpTcgjGuLU/uBL/ubcZF9OXbDo8ff4O8yVp5Bf0efS8uEoYo5q4Fx7dY9OgQGXgAsQA==",

"dev": true,

"license": "MIT"

},

"node\_modules/concat-map": {

"version": "0.0.1",

"resolved": "https://registry.npmjs.org/concat-map/-/concat-map-0.0.1.tgz",

"integrity": "sha512-/Srv4dswyQNBfohGpz9o6Yb3Gz3SrUDqBH5rTuhGR7ahtlbYKnVxw2bCFMRljaA7EXHaXZ8wsHdodFvbkhKmqg==",

"dev": true,

"license": "MIT"

},

"node\_modules/cross-spawn": {

"version": "7.0.6",

"resolved": "https://registry.npmjs.org/cross-spawn/-/cross-spawn-7.0.6.tgz",

"integrity": "sha512-uV2QOWP2nWzsy2aMp8aRibhi9dlzF5Hgh5SHaB9OiTGEyDTiJJyx0uy51QXdyWbtAHNua4XJzUKca3OzKUd3vA==",

"dev": true,

"license": "MIT",

"dependencies": {

"path-key": "^3.1.0",

"shebang-command": "^2.0.0",

"which": "^2.0.1"

},

"engines": {

"node": ">= 8"

}

},

"node\_modules/debug": {

"version": "4.4.1",

"resolved": "https://registry.npmjs.org/debug/-/debug-4.4.1.tgz",

"integrity": "sha512-KcKCqiftBJcZr++7ykoDIEwSa3XWowTfNPo92BYxjXiyYEVrUQh2aLyhxBCwww+heortUFxEJYcRzosstTEBYQ==",

"dev": true,

"license": "MIT",

"dependencies": {

"ms": "^2.1.3"

},

"engines": {

"node": ">=6.0"

},

"peerDependenciesMeta": {

"supports-color": {

"optional": true

}

}

},

"node\_modules/dotenv": {

"version": "16.5.0",

"resolved": "https://registry.npmjs.org/dotenv/-/dotenv-16.5.0.tgz",

"integrity": "sha512-m/C+AwOAr9/W1UOIZUo232ejMNnJAJtYQjUbHoNTBNTJSvqzzDh7vnrei3o3r3m9blf6ZoDkvcw0VmozNRFJxg==",

"license": "BSD-2-Clause",

"engines": {

"node": ">=12"

},

"funding": {

"url": "https://dotenvx.com"

}

},

"node\_modules/eastasianwidth": {

"version": "0.2.0",

"resolved": "https://registry.npmjs.org/eastasianwidth/-/eastasianwidth-0.2.0.tgz",

"integrity": "sha512-I88TYZWc9XiYHRQ4/3c5rjjfgkjhLyW2luGIheGERbNQ6OY7yTybanSpDXZa8y7VUP9YmDcYa+eyq4ca7iLqWA==",

"dev": true,

"license": "MIT"

},

"node\_modules/ecdsa-sig-formatter": {

"version": "1.0.11",

"resolved": "https://registry.npmjs.org/ecdsa-sig-formatter/-/ecdsa-sig-formatter-1.0.11.tgz",

"integrity": "sha512-nagl3RYrbNv6kQkeJIpt6NJZy8twLB/2vtz6yN9Z4vRKHN4/QZJIEbqohALSgwKdnksuY3k5Addp5lg8sVoVcQ==",

"license": "Apache-2.0",

"dependencies": {

"safe-buffer": "^5.0.1"

}

},

"node\_modules/emoji-regex": {

"version": "9.2.2",

"resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-regex-9.2.2.tgz",

"integrity": "sha512-L18DaJsXSUk2+42pv8mLs5jJT2hqFkFE4j21wOmgbUqsZ2hL72NsUU785g9RXgo3s0ZNgVl42TiHp3ZtOv/Vyg==",

"dev": true,

"license": "MIT"

},

"node\_modules/escalade": {

"version": "3.2.0",

"resolved": "https://registry.npmjs.org/escalade/-/escalade-3.2.0.tgz",

"integrity": "sha512-WUj2qlxaQtO4g6Pq5c29GTcWGDyd8itL8zTlipgECz3JesAiiOKotd8JU6otB3PACgG6xkJUyVhboMS+bje/jA==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=6"

}

},

"node\_modules/fill-range": {

"version": "7.1.1",

"resolved": "https://registry.npmjs.org/fill-range/-/fill-range-7.1.1.tgz",

"integrity": "sha512-YsGpe3WHLK8ZYi4tWDg2Jy3ebRz2rXowDxnld4bkQB00cc/1Zw9AWnC0i9ztDJitivtQvaI9KaLyKrc+hBW0yg==",

"dev": true,

"license": "MIT",

"dependencies": {

"to-regex-range": "^5.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/foreground-child": {

"version": "3.3.1",

"resolved": "https://registry.npmjs.org/foreground-child/-/foreground-child-3.3.1.tgz",

"integrity": "sha512-gIXjKqtFuWEgzFRJA9WCQeSJLZDjgJUOMCMzxtvFq/37KojM1BFGufqsCy0r4qSQmYLsZYMeyRqzIWOMup03sw==",

"dev": true,

"license": "ISC",

"dependencies": {

"cross-spawn": "^7.0.6",

"signal-exit": "^4.0.1"

},

"engines": {

"node": ">=14"

},

"funding": {

"url": "https://github.com/sponsors/isaacs"

}

},

"node\_modules/fsevents": {

"version": "2.3.3",

"resolved": "https://registry.npmjs.org/fsevents/-/fsevents-2.3.3.tgz",

"integrity": "sha512-5xoDfX+fL7faATnagmWPpbFtwh/R77WmMMqqHGS65C3vvB0YHrgF+B1YmZ3441tMj5n63k0212XNoJwzlhffQw==",

"dev": true,

"hasInstallScript": true,

"license": "MIT",

"optional": true,

"os": [

"darwin"

],

"engines": {

"node": "^8.16.0 || ^10.6.0 || >=11.0.0"

}

},

"node\_modules/get-caller-file": {

"version": "2.0.5",

"resolved": "https://registry.npmjs.org/get-caller-file/-/get-caller-file-2.0.5.tgz",

"integrity": "sha512-DyFP3BM/3YHTQOCUL/w0OZHR0lpKeGrxotcHWcqNEdnltqFwXVfhEBQ94eIo34AfQpo0rGki4cyIiftY06h2Fg==",

"dev": true,

"license": "ISC",

"engines": {

"node": "6.\* || 8.\* || >= 10.\*"

}

},

"node\_modules/glob": {

"version": "11.0.2",

"resolved": "https://registry.npmjs.org/glob/-/glob-11.0.2.tgz",

"integrity": "sha512-YT7U7Vye+t5fZ/QMkBFrTJ7ZQxInIUjwyAjVj84CYXqgBdv30MFUPGnBR6sQaVq6Is15wYJUsnzTuWaGRBhBAQ==",

"dev": true,

"license": "ISC",

"dependencies": {

"foreground-child": "^3.1.0",

"jackspeak": "^4.0.1",

"minimatch": "^10.0.0",

"minipass": "^7.1.2",

"package-json-from-dist": "^1.0.0",

"path-scurry": "^2.0.0"

},

"bin": {

"glob": "dist/esm/bin.mjs"

},

"engines": {

"node": "20 || >=22"

},

"funding": {

"url": "https://github.com/sponsors/isaacs"

}

},

"node\_modules/glob-parent": {

"version": "5.1.2",

"resolved": "https://registry.npmjs.org/glob-parent/-/glob-parent-5.1.2.tgz",

"integrity": "sha512-AOIgSQCepiJYwP3ARnGx+5VnTu2HBYdzbGP45eLw1vr3zB3vZLeyed1sC9hnbcOc9/SrMyM5RPQrkGz4aS9Zow==",

"dev": true,

"license": "ISC",

"dependencies": {

"is-glob": "^4.0.1"

},

"engines": {

"node": ">= 6"

}

},

"node\_modules/has-flag": {

"version": "3.0.0",

"resolved": "https://registry.npmjs.org/has-flag/-/has-flag-3.0.0.tgz",

"integrity": "sha512-sKJf1+ceQBr4SMkvQnBDNDtf4TXpVhVGateu0t918bl30FnbE2m4vNLX+VWe/dpjlb+HugGYzW7uQXH98HPEYw==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=4"

}

},

"node\_modules/ignore-by-default": {

"version": "1.0.1",

"resolved": "https://registry.npmjs.org/ignore-by-default/-/ignore-by-default-1.0.1.tgz",

"integrity": "sha512-Ius2VYcGNk7T90CppJqcIkS5ooHUZyIQK+ClZfMfMNFEF9VSE73Fq+906u/CWu92x4gzZMWOwfFYckPObzdEbA==",

"dev": true,

"license": "ISC"

},

"node\_modules/is-binary-path": {

"version": "2.1.0",

"resolved": "https://registry.npmjs.org/is-binary-path/-/is-binary-path-2.1.0.tgz",

"integrity": "sha512-ZMERYes6pDydyuGidse7OsHxtbI7WVeUEozgR/g7rd0xUimYNlvZRE/K2MgZTjWy725IfelLeVcEM97mmtRGXw==",

"dev": true,

"license": "MIT",

"dependencies": {

"binary-extensions": "^2.0.0"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/is-extglob": {

"version": "2.1.1",

"resolved": "https://registry.npmjs.org/is-extglob/-/is-extglob-2.1.1.tgz",

"integrity": "sha512-SbKbANkN603Vi4jEZv49LeVJMn4yGwsbzZworEoyEiutsN3nJYdbO36zfhGJ6QEDpOZIFkDtnq5JRxmvl3jsoQ==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/is-fullwidth-code-point": {

"version": "3.0.0",

"resolved": "https://registry.npmjs.org/is-fullwidth-code-point/-/is-fullwidth-code-point-3.0.0.tgz",

"integrity": "sha512-zymm5+u+sCsSWyD9qNaejV3DFvhCKclKdizYaJUuHA83RLjb7nSuGnddCHGv0hk+KY7BMAlsWeK4Ueg6EV6XQg==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/is-glob": {

"version": "4.0.3",

"resolved": "https://registry.npmjs.org/is-glob/-/is-glob-4.0.3.tgz",

"integrity": "sha512-xelSayHH36ZgE7ZWhli7pW34hNbNl8Ojv5KVmkJD4hBdD3th8Tfk9vYasLM+mXWOZhFkgZfxhLSnrwRr4elSSg==",

"dev": true,

"license": "MIT",

"dependencies": {

"is-extglob": "^2.1.1"

},

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/is-number": {

"version": "7.0.0",

"resolved": "https://registry.npmjs.org/is-number/-/is-number-7.0.0.tgz",

"integrity": "sha512-41Cifkg6e8TylSpdtTpeLVMqvSBEVzTttHvERD741+pnZ8ANv0004MRL43QKPDlK9cGvNp6NZWZUBlbGXYxxng==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=0.12.0"

}

},

"node\_modules/isexe": {

"version": "2.0.0",

"resolved": "https://registry.npmjs.org/isexe/-/isexe-2.0.0.tgz",

"integrity": "sha512-RHxMLp9lnKHGHRng9QFhRCMbYAcVpn69smSGcq3f36xjgVVWThj4qqLbTLlq7Ssj8B+fIQ1EuCEGI2lKsyQeIw==",

"dev": true,

"license": "ISC"

},

"node\_modules/jackspeak": {

"version": "4.1.1",

"resolved": "https://registry.npmjs.org/jackspeak/-/jackspeak-4.1.1.tgz",

"integrity": "sha512-zptv57P3GpL+O0I7VdMJNBZCu+BPHVQUk55Ft8/QCJjTVxrnJHuVuX/0Bl2A6/+2oyR/ZMEuFKwmzqqZ/U5nPQ==",

"dev": true,

"license": "BlueOak-1.0.0",

"dependencies": {

"@isaacs/cliui": "^8.0.2"

},

"engines": {

"node": "20 || >=22"

},

"funding": {

"url": "https://github.com/sponsors/isaacs"

}

},

"node\_modules/joi": {

"version": "17.13.3",

"resolved": "https://registry.npmjs.org/joi/-/joi-17.13.3.tgz",

"integrity": "sha512-otDA4ldcIx+ZXsKHWmp0YizCweVRZG96J10b0FevjfuncLO1oX59THoAmHkNubYJ+9gWsYsp5k8v4ib6oDv1fA==",

"license": "BSD-3-Clause",

"dependencies": {

"@hapi/hoek": "^9.3.0",

"@hapi/topo": "^5.1.0",

"@sideway/address": "^4.1.5",

"@sideway/formula": "^3.0.1",

"@sideway/pinpoint": "^2.0.0"

}

},

"node\_modules/jsonwebtoken": {

"version": "9.0.2",

"resolved": "https://registry.npmjs.org/jsonwebtoken/-/jsonwebtoken-9.0.2.tgz",

"integrity": "sha512-PRp66vJ865SSqOlgqS8hujT5U4AOgMfhrwYIuIhfKaoSCZcirrmASQr8CX7cUg+RMih+hgznrjp99o+W4pJLHQ==",

"license": "MIT",

"dependencies": {

"jws": "^3.2.2",

"lodash.includes": "^4.3.0",

"lodash.isboolean": "^3.0.3",

"lodash.isinteger": "^4.0.4",

"lodash.isnumber": "^3.0.3",

"lodash.isplainobject": "^4.0.6",

"lodash.isstring": "^4.0.1",

"lodash.once": "^4.0.0",

"ms": "^2.1.1",

"semver": "^7.5.4"

},

"engines": {

"node": ">=12",

"npm": ">=6"

}

},

"node\_modules/jwa": {

"version": "1.4.2",

"resolved": "https://registry.npmjs.org/jwa/-/jwa-1.4.2.tgz",

"integrity": "sha512-eeH5JO+21J78qMvTIDdBXidBd6nG2kZjg5Ohz/1fpa28Z4CcsWUzJ1ZZyFq/3z3N17aZy+ZuBoHljASbL1WfOw==",

"license": "MIT",

"dependencies": {

"buffer-equal-constant-time": "^1.0.1",

"ecdsa-sig-formatter": "1.0.11",

"safe-buffer": "^5.0.1"

}

},

"node\_modules/jws": {

"version": "3.2.2",

"resolved": "https://registry.npmjs.org/jws/-/jws-3.2.2.tgz",

"integrity": "sha512-YHlZCB6lMTllWDtSPHz/ZXTsi8S00usEV6v1tjq8tOUZzw7DpSDWVXjXDre6ed1w/pd495ODpHZYSdkRTsa0HA==",

"license": "MIT",

"dependencies": {

"jwa": "^1.4.1",

"safe-buffer": "^5.0.1"

}

},

"node\_modules/lodash.includes": {

"version": "4.3.0",

"resolved": "https://registry.npmjs.org/lodash.includes/-/lodash.includes-4.3.0.tgz",

"integrity": "sha512-W3Bx6mdkRTGtlJISOvVD/lbqjTlPPUDTMnlXZFnVwi9NKJ6tiAk6LVdlhZMm17VZisqhKcgzpO5Wz91PCt5b0w==",

"license": "MIT"

},

"node\_modules/lodash.isboolean": {

"version": "3.0.3",

"resolved": "https://registry.npmjs.org/lodash.isboolean/-/lodash.isboolean-3.0.3.tgz",

"integrity": "sha512-Bz5mupy2SVbPHURB98VAcw+aHh4vRV5IPNhILUCsOzRmsTmSQ17jIuqopAentWoehktxGd9e/hbIXq980/1QJg==",

"license": "MIT"

},

"node\_modules/lodash.isinteger": {

"version": "4.0.4",

"resolved": "https://registry.npmjs.org/lodash.isinteger/-/lodash.isinteger-4.0.4.tgz",

"integrity": "sha512-DBwtEWN2caHQ9/imiNeEA5ys1JoRtRfY3d7V9wkqtbycnAmTvRRmbHKDV4a0EYc678/dia0jrte4tjYwVBaZUA==",

"license": "MIT"

},

"node\_modules/lodash.isnumber": {

"version": "3.0.3",

"resolved": "https://registry.npmjs.org/lodash.isnumber/-/lodash.isnumber-3.0.3.tgz",

"integrity": "sha512-QYqzpfwO3/CWf3XP+Z+tkQsfaLL/EnUlXWVkIk5FUPc4sBdTehEqZONuyRt2P67PXAk+NXmTBcc97zw9t1FQrw==",

"license": "MIT"

},

"node\_modules/lodash.isplainobject": {

"version": "4.0.6",

"resolved": "https://registry.npmjs.org/lodash.isplainobject/-/lodash.isplainobject-4.0.6.tgz",

"integrity": "sha512-oSXzaWypCMHkPC3NvBEaPHf0KsA5mvPrOPgQWDsbg8n7orZ290M0BmC/jgRZ4vcJ6DTAhjrsSYgdsW/F+MFOBA==",

"license": "MIT"

},

"node\_modules/lodash.isstring": {

"version": "4.0.1",

"resolved": "https://registry.npmjs.org/lodash.isstring/-/lodash.isstring-4.0.1.tgz",

"integrity": "sha512-0wJxfxH1wgO3GrbuP+dTTk7op+6L41QCXbGINEmD+ny/G/eCqGzxyCsh7159S+mgDDcoarnBw6PC1PS5+wUGgw==",

"license": "MIT"

},

"node\_modules/lodash.once": {

"version": "4.1.1",

"resolved": "https://registry.npmjs.org/lodash.once/-/lodash.once-4.1.1.tgz",

"integrity": "sha512-Sb487aTOCr9drQVL8pIxOzVhafOjZN9UU54hiN8PU3uAiSV7lx1yYNpbNmex2PK6dSJoNTSJUUswT651yww3Mg==",

"license": "MIT"

},

"node\_modules/lru-cache": {

"version": "11.1.0",

"resolved": "https://registry.npmjs.org/lru-cache/-/lru-cache-11.1.0.tgz",

"integrity": "sha512-QIXZUBJUx+2zHUdQujWejBkcD9+cs94tLn0+YL8UrCh+D5sCXZ4c7LaEH48pNwRY3MLDgqUFyhlCyjJPf1WP0A==",

"dev": true,

"license": "ISC",

"engines": {

"node": "20 || >=22"

}

},

"node\_modules/mime-db": {

"version": "1.54.0",

"resolved": "https://registry.npmjs.org/mime-db/-/mime-db-1.54.0.tgz",

"integrity": "sha512-aU5EJuIN2WDemCcAp2vFBfp/m4EAhWJnUNSSw0ixs7/kXbd6Pg64EmwJkNdFhB8aWt1sH2CTXrLxo/iAGV3oPQ==",

"license": "MIT",

"engines": {

"node": ">= 0.6"

}

},

"node\_modules/minimatch": {

"version": "10.0.1",

"resolved": "https://registry.npmjs.org/minimatch/-/minimatch-10.0.1.tgz",

"integrity": "sha512-ethXTt3SGGR+95gudmqJ1eNhRO7eGEGIgYA9vnPatK4/etz2MEVDno5GMCibdMTuBMyElzIlgxMna3K94XDIDQ==",

"dev": true,

"license": "ISC",

"dependencies": {

"brace-expansion": "^2.0.1"

},

"engines": {

"node": "20 || >=22"

},

"funding": {

"url": "https://github.com/sponsors/isaacs"

}

},

"node\_modules/minipass": {

"version": "7.1.2",

"resolved": "https://registry.npmjs.org/minipass/-/minipass-7.1.2.tgz",

"integrity": "sha512-qOOzS1cBTWYF4BH8fVePDBOO9iptMnGUEZwNc/cMWnTV2nVLZ7VoNWEPHkYczZA0pdoA7dl6e7FL659nX9S2aw==",

"dev": true,

"license": "ISC",

"engines": {

"node": ">=16 || 14 >=14.17"

}

},

"node\_modules/ms": {

"version": "2.1.3",

"resolved": "https://registry.npmjs.org/ms/-/ms-2.1.3.tgz",

"integrity": "sha512-6FlzubTLZG3J2a/NVCAleEhjzq5oxgHyaCU9yYXvcLsvoVaHJq/s5xXI6/XXP6tz7R9xAOtHnSO/tXtF3WRTlA==",

"license": "MIT"

},

"node\_modules/nanoid": {

"version": "3.3.11",

"resolved": "https://registry.npmjs.org/nanoid/-/nanoid-3.3.11.tgz",

"integrity": "sha512-N8SpfPUnUp1bK+PMYW8qSWdl9U+wwNWI4QKxOYDy9JAro3WMX7p2OeVRF9v+347pnakNevPmiHhNmZ2HbFA76w==",

"funding": [

{

"type": "github",

"url": "https://github.com/sponsors/ai"

}

],

"license": "MIT",

"bin": {

"nanoid": "bin/nanoid.cjs"

},

"engines": {

"node": "^10 || ^12 || ^13.7 || ^14 || >=15.0.1"

}

},

"node\_modules/node-addon-api": {

"version": "8.4.0",

"resolved": "https://registry.npmjs.org/node-addon-api/-/node-addon-api-8.4.0.tgz",

"integrity": "sha512-D9DI/gXHvVmjHS08SVch0Em8G5S1P+QWtU31appcKT/8wFSPRcdHadIFSAntdMMVM5zz+/DL+bL/gz3UDppqtg==",

"license": "MIT",

"engines": {

"node": "^18 || ^20 || >= 21"

}

},

"node\_modules/node-gyp-build": {

"version": "4.8.4",

"resolved": "https://registry.npmjs.org/node-gyp-build/-/node-gyp-build-4.8.4.tgz",

"integrity": "sha512-LA4ZjwlnUblHVgq0oBF3Jl/6h/Nvs5fzBLwdEF4nuxnFdsfajde4WfxtJr3CaiH+F6ewcIB/q4jQ4UzPyid+CQ==",

"license": "MIT",

"bin": {

"node-gyp-build": "bin.js",

"node-gyp-build-optional": "optional.js",

"node-gyp-build-test": "build-test.js"

}

},

"node\_modules/node-pg-migrate": {

"version": "8.0.1",

"resolved": "https://registry.npmjs.org/node-pg-migrate/-/node-pg-migrate-8.0.1.tgz",

"integrity": "sha512-v0WL07EHt1x1VOBqYuYDbLSnbabEQGXNdyXf0H1Z+YQ5Zw8x+emI6wLLCl1Q7lfLYYV5291aWXdj8ixvkDCqBw==",

"dev": true,

"license": "MIT",

"dependencies": {

"glob": "~11.0.0",

"yargs": "~17.7.0"

},

"bin": {

"node-pg-migrate": "bin/node-pg-migrate.js"

},

"engines": {

"node": ">=20.11.0"

},

"peerDependencies": {

"@types/pg": ">=6.0.0 <9.0.0",

"pg": ">=4.3.0 <9.0.0"

},

"peerDependenciesMeta": {

"@types/pg": {

"optional": true

}

}

},

"node\_modules/nodemon": {

"version": "3.1.10",

"resolved": "https://registry.npmjs.org/nodemon/-/nodemon-3.1.10.tgz",

"integrity": "sha512-WDjw3pJ0/0jMFmyNDp3gvY2YizjLmmOUQo6DEBY+JgdvW/yQ9mEeSw6H5ythl5Ny2ytb7f9C2nIbjSxMNzbJXw==",

"dev": true,

"license": "MIT",

"dependencies": {

"chokidar": "^3.5.2",

"debug": "^4",

"ignore-by-default": "^1.0.1",

"minimatch": "^3.1.2",

"pstree.remy": "^1.1.8",

"semver": "^7.5.3",

"simple-update-notifier": "^2.0.0",

"supports-color": "^5.5.0",

"touch": "^3.1.0",

"undefsafe": "^2.0.5"

},

"bin": {

"nodemon": "bin/nodemon.js"

},

"engines": {

"node": ">=10"

},

"funding": {

"type": "opencollective",

"url": "https://opencollective.com/nodemon"

}

},

"node\_modules/nodemon/node\_modules/brace-expansion": {

"version": "1.1.11",

"resolved": "https://registry.npmjs.org/brace-expansion/-/brace-expansion-1.1.11.tgz",

"integrity": "sha512-iCuPHDFgrHX7H2vEI/5xpz07zSHB00TpugqhmYtVmMO6518mCuRMoOYFldEBl0g187ufozdaHgWKcYFb61qGiA==",

"dev": true,

"license": "MIT",

"dependencies": {

"balanced-match": "^1.0.0",

"concat-map": "0.0.1"

}

},

"node\_modules/nodemon/node\_modules/minimatch": {

"version": "3.1.2",

"resolved": "https://registry.npmjs.org/minimatch/-/minimatch-3.1.2.tgz",

"integrity": "sha512-J7p63hRiAjw1NDEww1W7i37+ByIrOWO5XQQAzZ3VOcL0PNybwpfmV/N05zFAzwQ9USyEcX6t3UO+K5aqBQOIHw==",

"dev": true,

"license": "ISC",

"dependencies": {

"brace-expansion": "^1.1.7"

},

"engines": {

"node": "\*"

}

},

"node\_modules/normalize-path": {

"version": "3.0.0",

"resolved": "https://registry.npmjs.org/normalize-path/-/normalize-path-3.0.0.tgz",

"integrity": "sha512-6eZs5Ls3WtCisHWp9S2GUy8dqkpGi4BVSz3GaqiE6ezub0512ESztXUwUB6C6IKbQkY2Pnb/mD4WYojCRwcwLA==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/package-json-from-dist": {

"version": "1.0.1",

"resolved": "https://registry.npmjs.org/package-json-from-dist/-/package-json-from-dist-1.0.1.tgz",

"integrity": "sha512-UEZIS3/by4OC8vL3P2dTXRETpebLI2NiI5vIrjaD/5UtrkFX/tNbwjTSRAGC/+7CAo2pIcBaRgWmcBBHcsaCIw==",

"dev": true,

"license": "BlueOak-1.0.0"

},

"node\_modules/path-key": {

"version": "3.1.1",

"resolved": "https://registry.npmjs.org/path-key/-/path-key-3.1.1.tgz",

"integrity": "sha512-ojmeN0qd+y0jszEtoY48r0Peq5dwMEkIlCOu6Q5f41lfkswXuKtYrhgoTpLnyIcHm24Uhqx+5Tqm2InSwLhE6Q==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/path-scurry": {

"version": "2.0.0",

"resolved": "https://registry.npmjs.org/path-scurry/-/path-scurry-2.0.0.tgz",

"integrity": "sha512-ypGJsmGtdXUOeM5u93TyeIEfEhM6s+ljAhrk5vAvSx8uyY/02OvrZnA0YNGUrPXfpJMgI1ODd3nwz8Npx4O4cg==",

"dev": true,

"license": "BlueOak-1.0.0",

"dependencies": {

"lru-cache": "^11.0.0",

"minipass": "^7.1.2"

},

"engines": {

"node": "20 || >=22"

},

"funding": {

"url": "https://github.com/sponsors/isaacs"

}

},

"node\_modules/pg": {

"version": "8.16.0",

"resolved": "https://registry.npmjs.org/pg/-/pg-8.16.0.tgz",

"integrity": "sha512-7SKfdvP8CTNXjMUzfcVTaI+TDzBEeaUnVwiVGZQD1Hh33Kpev7liQba9uLd4CfN8r9mCVsD0JIpq03+Unpz+kg==",

"license": "MIT",

"dependencies": {

"pg-connection-string": "^2.9.0",

"pg-pool": "^3.10.0",

"pg-protocol": "^1.10.0",

"pg-types": "2.2.0",

"pgpass": "1.0.5"

},

"engines": {

"node": ">= 8.0.0"

},

"optionalDependencies": {

"pg-cloudflare": "^1.2.5"

},

"peerDependencies": {

"pg-native": ">=3.0.1"

},

"peerDependenciesMeta": {

"pg-native": {

"optional": true

}

}

},

"node\_modules/pg-cloudflare": {

"version": "1.2.5",

"resolved": "https://registry.npmjs.org/pg-cloudflare/-/pg-cloudflare-1.2.5.tgz",

"integrity": "sha512-OOX22Vt0vOSRrdoUPKJ8Wi2OpE/o/h9T8X1s4qSkCedbNah9ei2W2765be8iMVxQUsvgT7zIAT2eIa9fs5+vtg==",

"license": "MIT",

"optional": true

},

"node\_modules/pg-connection-string": {

"version": "2.9.0",

"resolved": "https://registry.npmjs.org/pg-connection-string/-/pg-connection-string-2.9.0.tgz",

"integrity": "sha512-P2DEBKuvh5RClafLngkAuGe9OUlFV7ebu8w1kmaaOgPcpJd1RIFh7otETfI6hAR8YupOLFTY7nuvvIn7PLciUQ==",

"license": "MIT"

},

"node\_modules/pg-int8": {

"version": "1.0.1",

"resolved": "https://registry.npmjs.org/pg-int8/-/pg-int8-1.0.1.tgz",

"integrity": "sha512-WCtabS6t3c8SkpDBUlb1kjOs7l66xsGdKpIPZsg4wR+B3+u9UAum2odSsF9tnvxg80h4ZxLWMy4pRjOsFIqQpw==",

"license": "ISC",

"engines": {

"node": ">=4.0.0"

}

},

"node\_modules/pg-pool": {

"version": "3.10.0",

"resolved": "https://registry.npmjs.org/pg-pool/-/pg-pool-3.10.0.tgz",

"integrity": "sha512-DzZ26On4sQ0KmqnO34muPcmKbhrjmyiO4lCCR0VwEd7MjmiKf5NTg/6+apUEu0NF7ESa37CGzFxH513CoUmWnA==",

"license": "MIT",

"peerDependencies": {

"pg": ">=8.0"

}

},

"node\_modules/pg-protocol": {

"version": "1.10.0",

"resolved": "https://registry.npmjs.org/pg-protocol/-/pg-protocol-1.10.0.tgz",

"integrity": "sha512-IpdytjudNuLv8nhlHs/UrVBhU0e78J0oIS/0AVdTbWxSOkFUVdsHC/NrorO6nXsQNDTT1kzDSOMJubBQviX18Q==",

"license": "MIT"

},

"node\_modules/pg-types": {

"version": "2.2.0",

"resolved": "https://registry.npmjs.org/pg-types/-/pg-types-2.2.0.tgz",

"integrity": "sha512-qTAAlrEsl8s4OiEQY69wDvcMIdQN6wdz5ojQiOy6YRMuynxenON0O5oCpJI6lshc6scgAY8qvJ2On/p+CXY0GA==",

"license": "MIT",

"dependencies": {

"pg-int8": "1.0.1",

"postgres-array": "~2.0.0",

"postgres-bytea": "~1.0.0",

"postgres-date": "~1.0.4",

"postgres-interval": "^1.1.0"

},

"engines": {

"node": ">=4"

}

},

"node\_modules/pgpass": {

"version": "1.0.5",

"resolved": "https://registry.npmjs.org/pgpass/-/pgpass-1.0.5.tgz",

"integrity": "sha512-FdW9r/jQZhSeohs1Z3sI1yxFQNFvMcnmfuj4WBMUTxOrAyLMaTcE1aAMBiTlbMNaXvBCQuVi0R7hd8udDSP7ug==",

"license": "MIT",

"dependencies": {

"split2": "^4.1.0"

}

},

"node\_modules/picomatch": {

"version": "2.3.1",

"resolved": "https://registry.npmjs.org/picomatch/-/picomatch-2.3.1.tgz",

"integrity": "sha512-JU3teHTNjmE2VCGFzuY8EXzCDVwEqB2a8fsIvwaStHhAWJEeVd1o1QD80CU6+ZdEXXSLbSsuLwJjkCBWqRQUVA==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8.6"

},

"funding": {

"url": "https://github.com/sponsors/jonschlinkert"

}

},

"node\_modules/postgres-array": {

"version": "2.0.0",

"resolved": "https://registry.npmjs.org/postgres-array/-/postgres-array-2.0.0.tgz",

"integrity": "sha512-VpZrUqU5A69eQyW2c5CA1jtLecCsN2U/bD6VilrFDWq5+5UIEVO7nazS3TEcHf1zuPYO/sqGvUvW62g86RXZuA==",

"license": "MIT",

"engines": {

"node": ">=4"

}

},

"node\_modules/postgres-bytea": {

"version": "1.0.0",

"resolved": "https://registry.npmjs.org/postgres-bytea/-/postgres-bytea-1.0.0.tgz",

"integrity": "sha512-xy3pmLuQqRBZBXDULy7KbaitYqLcmxigw14Q5sj8QBVLqEwXfeybIKVWiqAXTlcvdvb0+xkOtDbfQMOf4lST1w==",

"license": "MIT",

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/postgres-date": {

"version": "1.0.7",

"resolved": "https://registry.npmjs.org/postgres-date/-/postgres-date-1.0.7.tgz",

"integrity": "sha512-suDmjLVQg78nMK2UZ454hAG+OAW+HQPZ6n++TNDUX+L0+uUlLywnoxJKDou51Zm+zTCjrCl0Nq6J9C5hP9vK/Q==",

"license": "MIT",

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/postgres-interval": {

"version": "1.2.0",

"resolved": "https://registry.npmjs.org/postgres-interval/-/postgres-interval-1.2.0.tgz",

"integrity": "sha512-9ZhXKM/rw350N1ovuWHbGxnGh/SNJ4cnxHiM0rxE4VN41wsg8P8zWn9hv/buK00RP4WvlOyr/RBDiptyxVbkZQ==",

"license": "MIT",

"dependencies": {

"xtend": "^4.0.0"

},

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/pstree.remy": {

"version": "1.1.8",

"resolved": "https://registry.npmjs.org/pstree.remy/-/pstree.remy-1.1.8.tgz",

"integrity": "sha512-77DZwxQmxKnu3aR542U+X8FypNzbfJ+C5XQDk3uWjWxn6151aIMGthWYRXTqT1E5oJvg+ljaa2OJi+VfvCOQ8w==",

"dev": true,

"license": "MIT"

},

"node\_modules/readdirp": {

"version": "3.6.0",

"resolved": "https://registry.npmjs.org/readdirp/-/readdirp-3.6.0.tgz",

"integrity": "sha512-hOS089on8RduqdbhvQ5Z37A0ESjsqz6qnRcffsMU3495FuTdqSm+7bhJ29JvIOsBDEEnan5DPu9t3To9VRlMzA==",

"dev": true,

"license": "MIT",

"dependencies": {

"picomatch": "^2.2.1"

},

"engines": {

"node": ">=8.10.0"

}

},

"node\_modules/require-directory": {

"version": "2.1.1",

"resolved": "https://registry.npmjs.org/require-directory/-/require-directory-2.1.1.tgz",

"integrity": "sha512-fGxEI7+wsG9xrvdjsrlmL22OMTTiHRwAMroiEeMgq8gzoLC/PQr7RsRDSTLUg/bZAZtF+TVIkHc6/4RIKrui+Q==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=0.10.0"

}

},

"node\_modules/safe-buffer": {

"version": "5.2.1",

"resolved": "https://registry.npmjs.org/safe-buffer/-/safe-buffer-5.2.1.tgz",

"integrity": "sha512-rp3So07KcdmmKbGvgaNxQSJr7bGVSVk5S9Eq1F+ppbRo70+YeaDxkw5Dd8NPN+GD6bjnYm2VuPuCXmpuYvmCXQ==",

"funding": [

{

"type": "github",

"url": "https://github.com/sponsors/feross"

},

{

"type": "patreon",

"url": "https://www.patreon.com/feross"

},

{

"type": "consulting",

"url": "https://feross.org/support"

}

],

"license": "MIT"

},

"node\_modules/semver": {

"version": "7.7.2",

"resolved": "https://registry.npmjs.org/semver/-/semver-7.7.2.tgz",

"integrity": "sha512-RF0Fw+rO5AMf9MAyaRXI4AV0Ulj5lMHqVxxdSgiVbixSCXoEmmX/jk0CuJw4+3SqroYO9VoUh+HcuJivvtJemA==",

"license": "ISC",

"bin": {

"semver": "bin/semver.js"

},

"engines": {

"node": ">=10"

}

},

"node\_modules/shebang-command": {

"version": "2.0.0",

"resolved": "https://registry.npmjs.org/shebang-command/-/shebang-command-2.0.0.tgz",

"integrity": "sha512-kHxr2zZpYtdmrN1qDjrrX/Z1rR1kG8Dx+gkpK1G4eXmvXswmcE1hTWBWYUzlraYw1/yZp6YuDY77YtvbN0dmDA==",

"dev": true,

"license": "MIT",

"dependencies": {

"shebang-regex": "^3.0.0"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/shebang-regex": {

"version": "3.0.0",

"resolved": "https://registry.npmjs.org/shebang-regex/-/shebang-regex-3.0.0.tgz",

"integrity": "sha512-7++dFhtcx3353uBaq8DDR4NuxBetBzC7ZQOhmTQInHEd6bSrXdiEyzCvG07Z44UYdLShWUyXt5M/yhz8ekcb1A==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/signal-exit": {

"version": "4.1.0",

"resolved": "https://registry.npmjs.org/signal-exit/-/signal-exit-4.1.0.tgz",

"integrity": "sha512-bzyZ1e88w9O1iNJbKnOlvYTrWPDl46O1bG0D3XInv+9tkPrxrN8jUUTiFlDkkmKWgn1M6CfIA13SuGqOa9Korw==",

"dev": true,

"license": "ISC",

"engines": {

"node": ">=14"

},

"funding": {

"url": "https://github.com/sponsors/isaacs"

}

},

"node\_modules/simple-update-notifier": {

"version": "2.0.0",

"resolved": "https://registry.npmjs.org/simple-update-notifier/-/simple-update-notifier-2.0.0.tgz",

"integrity": "sha512-a2B9Y0KlNXl9u/vsW6sTIu9vGEpfKu2wRV6l1H3XEas/0gUIzGzBoP/IouTcUQbm9JWZLH3COxyn03TYlFax6w==",

"dev": true,

"license": "MIT",

"dependencies": {

"semver": "^7.5.3"

},

"engines": {

"node": ">=10"

}

},

"node\_modules/split2": {

"version": "4.2.0",

"resolved": "https://registry.npmjs.org/split2/-/split2-4.2.0.tgz",

"integrity": "sha512-UcjcJOWknrNkF6PLX83qcHM6KHgVKNkV62Y8a5uYDVv9ydGQVwAHMKqHdJje1VTWpljG0WYpCDhrCdAOYH4TWg==",

"license": "ISC",

"engines": {

"node": ">= 10.x"

}

},

"node\_modules/string-width": {

"version": "5.1.2",

"resolved": "https://registry.npmjs.org/string-width/-/string-width-5.1.2.tgz",

"integrity": "sha512-HnLOCR3vjcY8beoNLtcjZ5/nxn2afmME6lhrDrebokqMap+XbeW8n9TXpPDOqdGK5qcI3oT0GKTW6wC7EMiVqA==",

"dev": true,

"license": "MIT",

"dependencies": {

"eastasianwidth": "^0.2.0",

"emoji-regex": "^9.2.2",

"strip-ansi": "^7.0.1"

},

"engines": {

"node": ">=12"

},

"funding": {

"url": "https://github.com/sponsors/sindresorhus"

}

},

"node\_modules/string-width-cjs": {

"name": "string-width",

"version": "4.2.3",

"resolved": "https://registry.npmjs.org/string-width/-/string-width-4.2.3.tgz",

"integrity": "sha512-wKyQRQpjJ0sIp62ErSZdGsjMJWsap5oRNihHhu6G7JVO/9jIB6UyevL+tXuOqrng8j/cxKTWyWUwvSTriiZz/g==",

"dev": true,

"license": "MIT",

"dependencies": {

"emoji-regex": "^8.0.0",

"is-fullwidth-code-point": "^3.0.0",

"strip-ansi": "^6.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/string-width-cjs/node\_modules/ansi-regex": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.1.tgz",

"integrity": "sha512-quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMWv37qOOb9pdJlMUEKFQ==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/string-width-cjs/node\_modules/emoji-regex": {

"version": "8.0.0",

"resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-regex-8.0.0.tgz",

"integrity": "sha512-MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8Eu3TdM14SawnVUmGE6A==",

"dev": true,

"license": "MIT"

},

"node\_modules/string-width-cjs/node\_modules/strip-ansi": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-6.0.1.tgz",

"integrity": "sha512-Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/0ooI7KrPuUSztUdU5A==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-regex": "^5.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/strip-ansi": {

"version": "7.1.0",

"resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-7.1.0.tgz",

"integrity": "sha512-iq6eVVI64nQQTRYq2KtEg2d2uU7LElhTJwsH4YzIHZshxlgZms/wIc4VoDQTlG/IvVIrBKG06CrZnp0qv7hkcQ==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-regex": "^6.0.1"

},

"engines": {

"node": ">=12"

},

"funding": {

"url": "https://github.com/chalk/strip-ansi?sponsor=1"

}

},

"node\_modules/strip-ansi-cjs": {

"name": "strip-ansi",

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-6.0.1.tgz",

"integrity": "sha512-Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/0ooI7KrPuUSztUdU5A==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-regex": "^5.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/strip-ansi-cjs/node\_modules/ansi-regex": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.1.tgz",

"integrity": "sha512-quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMWv37qOOb9pdJlMUEKFQ==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/supports-color": {

"version": "5.5.0",

"resolved": "https://registry.npmjs.org/supports-color/-/supports-color-5.5.0.tgz",

"integrity": "sha512-QjVjwdXIt408MIiAqCX4oUKsgU2EqAGzs2Ppkm4aQYbjm+ZEWEcW4SfFNTr4uMNZma0ey4f5lgLrkB0aX0QMow==",

"dev": true,

"license": "MIT",

"dependencies": {

"has-flag": "^3.0.0"

},

"engines": {

"node": ">=4"

}

},

"node\_modules/to-regex-range": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/to-regex-range/-/to-regex-range-5.0.1.tgz",

"integrity": "sha512-65P7iz6X5yEr1cwcgvQxbbIw7Uk3gOy5dIdtZ4rDveLqhrdJP+Li/Hx6tyK0NEb+2GCyneCMJiGqrADCSNk8sQ==",

"dev": true,

"license": "MIT",

"dependencies": {

"is-number": "^7.0.0"

},

"engines": {

"node": ">=8.0"

}

},

"node\_modules/touch": {

"version": "3.1.1",

"resolved": "https://registry.npmjs.org/touch/-/touch-3.1.1.tgz",

"integrity": "sha512-r0eojU4bI8MnHr8c5bNo7lJDdI2qXlWWJk6a9EAFG7vbhTjElYhBVS3/miuE0uOuoLdb8Mc/rVfsmm6eo5o9GA==",

"dev": true,

"license": "ISC",

"bin": {

"nodetouch": "bin/nodetouch.js"

}

},

"node\_modules/undefsafe": {

"version": "2.0.5",

"resolved": "https://registry.npmjs.org/undefsafe/-/undefsafe-2.0.5.tgz",

"integrity": "sha512-WxONCrssBM8TSPRqN5EmsjVrsv4A8X12J4ArBiiayv3DyyG3ZlIg6yysuuSYdZsVz3TKcTg2fd//Ujd4CHV1iA==",

"dev": true,

"license": "MIT"

},

"node\_modules/which": {

"version": "2.0.2",

"resolved": "https://registry.npmjs.org/which/-/which-2.0.2.tgz",

"integrity": "sha512-BLI3Tl1TW3Pvl70l3yq3Y64i+awpwXqsGBYWkkqMtnbXgrMD+yj7rhW0kuEDxzJaYXGjEW5ogapKNMEKNMjibA==",

"dev": true,

"license": "ISC",

"dependencies": {

"isexe": "^2.0.0"

},

"bin": {

"node-which": "bin/node-which"

},

"engines": {

"node": ">= 8"

}

},

"node\_modules/wrap-ansi": {

"version": "8.1.0",

"resolved": "https://registry.npmjs.org/wrap-ansi/-/wrap-ansi-8.1.0.tgz",

"integrity": "sha512-si7QWI6zUMq56bESFvagtmzMdGOtoxfR+Sez11Mobfc7tm+VkUckk9bW2UeffTGVUbOksxmSw0AA2gs8g71NCQ==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-styles": "^6.1.0",

"string-width": "^5.0.1",

"strip-ansi": "^7.0.1"

},

"engines": {

"node": ">=12"

},

"funding": {

"url": "https://github.com/chalk/wrap-ansi?sponsor=1"

}

},

"node\_modules/wrap-ansi-cjs": {

"name": "wrap-ansi",

"version": "7.0.0",

"resolved": "https://registry.npmjs.org/wrap-ansi/-/wrap-ansi-7.0.0.tgz",

"integrity": "sha512-YVGIj2kamLSTxw6NsZjoBxfSwsn0ycdesmc4p+Q21c5zPuZ1pl+NfxVdxPtdHvmNVOQ6XSYG4AUtyt/Fi7D16Q==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-styles": "^4.0.0",

"string-width": "^4.1.0",

"strip-ansi": "^6.0.0"

},

"engines": {

"node": ">=10"

},

"funding": {

"url": "https://github.com/chalk/wrap-ansi?sponsor=1"

}

},

"node\_modules/wrap-ansi-cjs/node\_modules/ansi-regex": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.1.tgz",

"integrity": "sha512-quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMWv37qOOb9pdJlMUEKFQ==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/wrap-ansi-cjs/node\_modules/ansi-styles": {

"version": "4.3.0",

"resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-styles-4.3.0.tgz",

"integrity": "sha512-zbB9rCJAT1rbjiVDb2hqKFHNYLxgtk8NURxZ3IZwD3F6NtxbXZQCnnSi1Lkx+IDohdPlFp222wVALIheZJQSEg==",

"dev": true,

"license": "MIT",

"dependencies": {

"color-convert": "^2.0.1"

},

"engines": {

"node": ">=8"

},

"funding": {

"url": "https://github.com/chalk/ansi-styles?sponsor=1"

}

},

"node\_modules/wrap-ansi-cjs/node\_modules/emoji-regex": {

"version": "8.0.0",

"resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-regex-8.0.0.tgz",

"integrity": "sha512-MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8Eu3TdM14SawnVUmGE6A==",

"dev": true,

"license": "MIT"

},

"node\_modules/wrap-ansi-cjs/node\_modules/string-width": {

"version": "4.2.3",

"resolved": "https://registry.npmjs.org/string-width/-/string-width-4.2.3.tgz",

"integrity": "sha512-wKyQRQpjJ0sIp62ErSZdGsjMJWsap5oRNihHhu6G7JVO/9jIB6UyevL+tXuOqrng8j/cxKTWyWUwvSTriiZz/g==",

"dev": true,

"license": "MIT",

"dependencies": {

"emoji-regex": "^8.0.0",

"is-fullwidth-code-point": "^3.0.0",

"strip-ansi": "^6.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/wrap-ansi-cjs/node\_modules/strip-ansi": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-6.0.1.tgz",

"integrity": "sha512-Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/0ooI7KrPuUSztUdU5A==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-regex": "^5.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/xtend": {

"version": "4.0.2",

"resolved": "https://registry.npmjs.org/xtend/-/xtend-4.0.2.tgz",

"integrity": "sha512-LKYU1iAXJXUgAXn9URjiu+MWhyUXHsvfp7mcuYm9dSUKK0/CjtrUwFAxD82/mCWbtLsGjFIad0wIsod4zrTAEQ==",

"license": "MIT",

"engines": {

"node": ">=0.4"

}

},

"node\_modules/y18n": {

"version": "5.0.8",

"resolved": "https://registry.npmjs.org/y18n/-/y18n-5.0.8.tgz",

"integrity": "sha512-0pfFzegeDWJHJIAmTLRP2DwHjdF5s7jo9tuztdQxAhINCdvS+3nGINqPd00AphqJR/0LhANUS6/+7SCb98YOfA==",

"dev": true,

"license": "ISC",

"engines": {

"node": ">=10"

}

},

"node\_modules/yargs": {

"version": "17.7.2",

"resolved": "https://registry.npmjs.org/yargs/-/yargs-17.7.2.tgz",

"integrity": "sha512-7dSzzRQ++CKnNI/krKnYRV7JKKPUXMEh61soaHKg9mrWEhzFWhFnxPxGl+69cD1Ou63C13NUPCnmIcrvqCuM6w==",

"dev": true,

"license": "MIT",

"dependencies": {

"cliui": "^8.0.1",

"escalade": "^3.1.1",

"get-caller-file": "^2.0.5",

"require-directory": "^2.1.1",

"string-width": "^4.2.3",

"y18n": "^5.0.5",

"yargs-parser": "^21.1.1"

},

"engines": {

"node": ">=12"

}

},

"node\_modules/yargs-parser": {

"version": "21.1.1",

"resolved": "https://registry.npmjs.org/yargs-parser/-/yargs-parser-21.1.1.tgz",

"integrity": "sha512-tVpsJW7DdjecAiFpbIB1e3qxIQsE6NoPc5/eTdrbbIC4h0LVsWhnoa3g+m2HclBIujHzsxZ4VJVA+GUuc2/LBw==",

"dev": true,

"license": "ISC",

"engines": {

"node": ">=12"

}

},

"node\_modules/yargs/node\_modules/ansi-regex": {

"version": "5.0.1",

"resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.1.tgz",

"integrity": "sha512-quJQXlTSUGL2LH9SUXo8VwsY4soanhgo6LNSm84E1LBcE8s3O0wpdiRzyR9z/ZZJMlMWv37qOOb9pdJlMUEKFQ==",

"dev": true,

"license": "MIT",

"engines": {

"node": ">=8"

}

},

"node\_modules/yargs/node\_modules/emoji-regex": {

"version": "8.0.0",

"resolved": "https://registry.npmjs.org/emoji-regex/-/emoji-regex-8.0.0.tgz",

"integrity": "sha512-MSjYzcWNOA0ewAHpz0MxpYFvwg6yjy1NG3xteoqz644VCo/RPgnr1/GGt+ic3iJTzQ8Eu3TdM14SawnVUmGE6A==",

"dev": true,

"license": "MIT"

},

"node\_modules/yargs/node\_modules/string-width": {

"version": "4.2.3",

"resolved": "https://registry.npmjs.org/string-width/-/string-width-4.2.3.tgz",

"integrity": "sha512-wKyQRQpjJ0sIp62ErSZdGsjMJWsap5oRNihHhu6G7JVO/9jIB6UyevL+tXuOqrng8j/cxKTWyWUwvSTriiZz/g==",

"dev": true,

"license": "MIT",

"dependencies": {

"emoji-regex": "^8.0.0",

"is-fullwidth-code-point": "^3.0.0",

"strip-ansi": "^6.0.1"

},

"engines": {

"node": ">=8"

}

},

"node\_modules/yargs/node\_modules/strip-ansi": {

"version": "6.0.1",

"resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-6.0.1.tgz",

"integrity": "sha512-Y38VPSHcqkFrCpFnQ9vuSXmquuv5oXOKpGeT6aGrr3o3Gc9AlVa6JBfUSOCnbxGGZF+/0ooI7KrPuUSztUdU5A==",

"dev": true,

"license": "MIT",

"dependencies": {

"ansi-regex": "^5.0.1"

},

"engines": {

"node": ">=8"

}

}

}

}

{} package.json

{

  "name": "openmusic-api",

  "version": "1.0.0",

  "description": "",

  "main": "index.js",

  "scripts": {

    "start": "node src/server.js",

    "dev": "nodemon src/server.js",

    "migrate": "node-pg-migrate",

    "migrate:down": "node-pg-migrate down",

    "migrate:up": "node-pg-migrate up"

  },

  "keywords": [],

  "author": "",

  "license": "ISC",

  "type": "commonjs",

  "dependencies": {

    "@hapi/hapi": "^21.4.0",

    "@hapi/jwt": "^3.2.0",

    "auto-bind": "^4.0.0",

    "bcrypt": "^6.0.0",

    "dotenv": "^16.5.0",

    "joi": "^17.13.3",

    "jsonwebtoken": "^9.0.2",

    "nanoid": "^3.3.11",

    "pg": "^8.16.0"

  },

  "devDependencies": {

    "node-pg-migrate": "^8.0.1",

    "nodemon": "^3.1.10"

  }

}