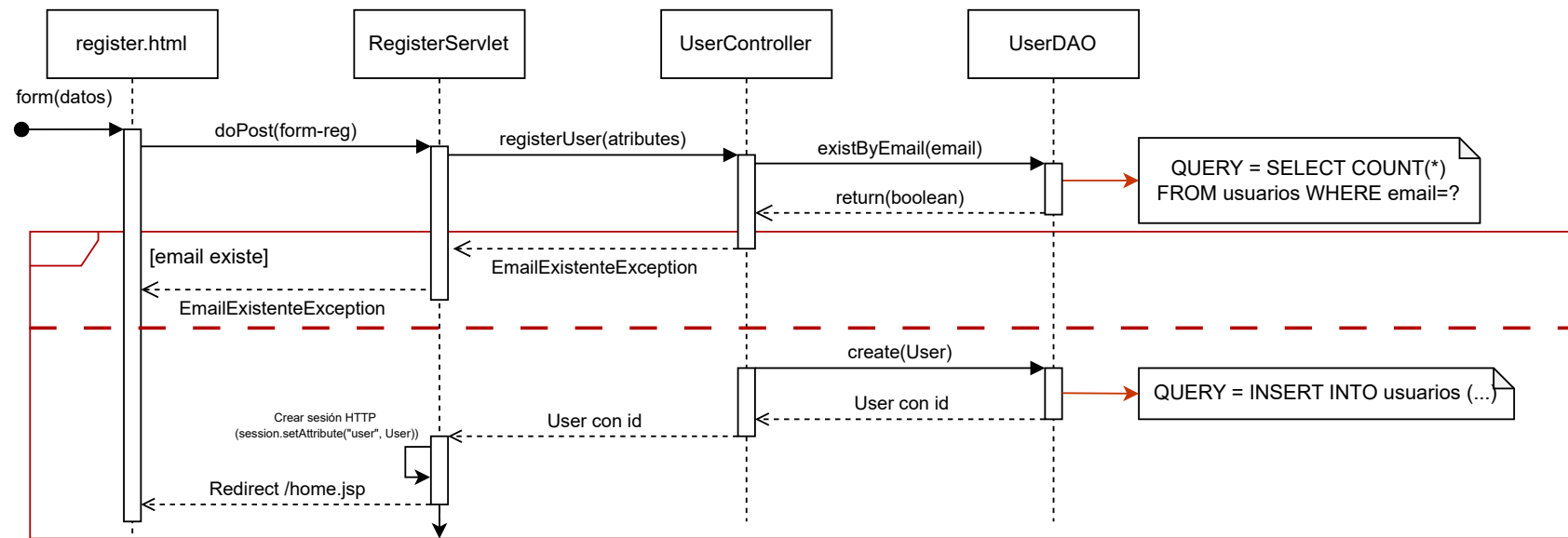


C.U. 01 "Registrar Usuario"



Contrato rápido

- Entrada (POST): email, password, confirmPassword, nombre, apellido, factionId
- Salida (éxito): Redirect a /dashboard con sesión HTTP "user"
- Salida (error): Forward a register.jsp con mensaje de error y datos del formulario
- Errores típicos: Email inválido, passwords no coinciden, email existente, facción inexistente, SQLException

Pasos y responsabilidades

- register.jsp
 - Muestra el formulario y envía POST a /register
- RegisterServlet.doPost(...)
 - Lee parámetros y hace validaciones básicas (formato, campos, password==confirm)
 - Construye UserRegistrationDTO y llama C.registerUser(dto)
 - Si éxito: crea sesión y hace redirect
 - Si error: setea atributos de error y hace forward a register.jsp
- UserController.registerUser(dto)
 - Verifica email único (UserDAO.existsByEmail)
 - Valida facción (FactionDAO.existsById) si ya lo tienes; si no, carga y valida lista válida
 - Hashea password con BCrypt
 - Setea defaults (tipo_usuario=PASAJERO, activo=true, verificado=false)
 - Llama UserDAO.create(User) y retorna User
- UserDAO
 - existsByEmail(email): SELECT COUNT(*)
 - create(User): INSERT ...; retorna User con id generado

Alternativas resumidas

- Email inválido/passwords no coinciden: Servlet retorna a JSP con error
- Email ya existe: Controller lanza EmailExistenteException → Servlet forward con error
- Facción no válida: Controller lanza FactionInvalidaException → Servlet forward con error
- Error DB (INSERT): Servlet muestra "Error interno, intente nuevamente"