Kronos

https://kronos/gerenciarFrequencia.html

Gerenciar Frequência

Curso RN001 | Periodo RN002 | Turma RN003 | Disciplina RN004 | Sub-Fase (Bimestre) RN005

RN006 Atividades	Registrar Realização
[Data Prevista] [Da Semana] [Data Realização]	[Confirmar] [Editar] [Visualizar]
[Data Prevista] [Da Semana] [Data Realização]	[Confirmar] [Editar] [Visualizar]

RN008 (Issue #107)

Frequência

RN013 [TP] [TF] [L]Impor]

Alunos RN007	Frequência	Ações
[Avatar (Foto)] [Nome] [Nr Atividade] [Data Reposição] [Tipo Frequência]	[P] [AO] [F] [FJ] [T]	[TF] [TF] [L]Impor] [Reposição]
[Avatar (Foto)] [Nome] [Nr Atividade] [Data Reposição] [Tipo Frequência]	[P] [AO] [F] [FJ] [T]	[TF] [AO] [L]Impor] [Reposição]

RN010 (Issue #108) RN011 (Issue #108)

Alterar Atividade RN009

Data Prevista | Dia Semana | Regime Letivo

Data Realização | Bimestre | Professor

Horas Atividades

HA 07:30 - 08:20 | HA 08:20 - 09:10 | HA 09:10 - 10:00

Cancelar (-1) | Alterar (-1) RN008 (Issue #107)

Reposição

Data da Reposição

Cancelar (-1) | Confirmar (-1)

Visualizar Atividade RN009

Data Prevista | Dia Semana | Regime Letivo

Data Realização | Bimestre | Professor

Horas Atividades

HA 07:30 - 08:20 | HA 08:20 - 09:10 | HA 09:10 - 10:00

Fechar (-1)

RN001
select distinct c.id, c.id_nivel, c.nome
from curso c
left join periodo p on (c.id = p.id_curso)
left join disciplina d on (p.id = d.id_periodo)
left join horario h on (d.id = h.id_disciplina)
left join funcionario f on (h.id_funcionario = f.id)
left join turma t on (h.id_turma = t.id)
where f.encerrada = false
and c.id_instituicao = ((instituicaoUsuarioLogado
and f.id_pessoa = ((pessoaUsuarioLogado

RN002
select distinct pe.id, p.id_faixa_ano, p.nome
from periodo_execucao pe
left join turma t on (pe.id = t.id_periodo_execucao)
left join horario h on (d.id = h.id_turma)
left join funcionario f on (h.id_funcionario = f.id)
left join periodo p on (pe.id_periodo = p.id)
where p.id_curso = 4 = ((CursoSelecionado
and f.id_pessoa = 4 --idPessoaUsuarioLogado

RN003
select distinct t.id, t.ano, t.sigla//'/'||t.sigla as turma
from turma t
left join tipo_turno tt on (t.id_tipo_turno = tt.id)
left join horario h on (d.id = h.id_turma)
left join funcionario f on (h.id_funcionario = f.id)
where f.id_periodo_execucao = 20 -- idPeriodoExecucaoSelecionado
and f.id_pessoa = 2 --idPessoaUsuarioLogado

RN004
select distinct d.id, d.sigla
from horario h
left join disciplina d on (h.id_disciplina = d.id)
left join funcionario f on (h.id_funcionario = f.id)
where h.id_turma = 20 -- idTurmaSelecionada
and f.id_pessoa = 2 --idPessoaUsuarioLogado

RN005
select distinct sfe.id, sf.sigla
from sub_fase_execucao sfe
left join sub_fase sf on (sfe.id_sub_fase = sf.id)
left join fase_execucao fe on (sfe.id_fase_execucao = fe.id)
left join periodo_execucao pe on (fe.id_periodo_execucao = pe.id)
left join turma t on (pe.id = t.id_periodo_execucao)
left join horario h on (t.id = h.id_turma)
left join funcionario f on (h.id_funcionario = f.id)
where h.id_turma = 20 -- idTurmaSelecionada
and f.id_pessoa = 2 --idPessoaUsuarioLogado
and h.id_disciplina = 50 --idDisciplinaSelecionada

RN006
-- Para cada Sub_Fase (Bimestre) selecionado:
select
a.id, a.data_prevista, a.data_realizacao, t.sigla
from atividade a
left join horario h on (a.id_horario = h.id)
left join funcionario f on (h.id_funcionario = f.id)
left join tipo_dia_semana t on (h.id_tipo_dia_semana = t.id)
where a.id_sub_fase = 53 --idSubFaseSelecionada
and h.id_disciplina = 53 --idDisciplinaSelecionada
and a.id_sub_fase_execucao = 60 --idSubFaseExecucaoSelecionada
order by a.data_prevista

RN007
-- Para cada seleção de atividade:
select
f.id, a.id as id_atividade, p.id as id_pessoa, c.id as id_credito, p.nome, f.data_reposicao,
count(ha.id_hora_atividade) as nrAtividade,
cast((select sum(f.nr_falta)
from atividade ar
left join frequencia f on (ar.id = f.id_atividade)
left join horario ho on (ar.id_horario = ho.id)
where ar.id_sub_fase_execucao = 60 --idSubFaseExecucaoSelecionada
and ho.id_turma = 22 --idTurmaSelecionada
and ho.id_disciplina = 55 --idDisciplinaSelecionada
and f.id_credito = c.id)
as integer) as NrFaltas,
CASE
WHEN (select tf.sigla
from atestado at
left join tipo_falta tf on (at.id_tipo_falta = tf.id)
where at.id_tipo_atestado = 2
and at.id_pessoa = p.id
and a.data_prevista between at.data_inicio_vigencia and at.data_final_vigencia) is not null THEN 'T'
WHEN (select tf.sigla
from atestado at
left join tipo_falta tf on (at.id_tipo_falta = tf.id)
where at.id_tipo_atestado = 4
and at.id_pessoa = p.id
and a.data_prevista between at.data_inicio_vigencia and at.data_final_vigencia) is not null THEN 'C'
WHEN (select tf.sigla
from atestado at
left join tipo_falta tf on (at.id_tipo_falta = tf.id)
where at.id_tipo_atestado = 3
and a.data_prevista between at.data_inicio_vigencia and at.data_final_vigencia) is not null THEN 'AD'
WHEN (select tf.sigla
from atestado at
left join tipo_falta tf on (at.id_tipo_falta = tf.id)
where at.id_tipo_atestado = 1
and at.id_pessoa = p.id
and a.data_prevista between at.data_inicio_vigencia and at.data_final_vigencia) is not null THEN
(select tf.sigla
from atestado at
left join tipo_falta tf on (at.id_tipo_falta = tf.id)
where at.id_tipo_atestado = 2
and at.id_pessoa = p.id
and a.data_prevista between at.data_inicio_vigencia and at.data_final_vigencia)
ELSE
null
END AS tp_frequencia,
f.frequencia
from atividade a
left join horario h on (a.id_horario = h.id)
left join turma t on (h.id_turma = t.id)
left join credito c on (t.id = c.id_turma)
left join contrato ct on (c.id_contrato = ct.id)
left join matricula m on (ct.id_matricula = m.id)
left join pessoa p on (m.id_pessoa = p.id)
left join frequencia f on (a.id = f.id_atividade and c.id = f.id_credito)
left join horario_hora_atividade ha on (h.id = ha.id_horario)
where a.id = 2 --IdAtividadeSelecionada
and h.id_disciplina = c.id_disciplina
group by a.data_prevista, f.id, a.id, p.id, c.id, p.nome, f.data_reposicao, f.frequencia
order by p.nome

RN008
Ao acionar a ação "Confirmar" o sistema deverá:

1. Executar a verificação conforme os seguintes sql:

select CASE
WHEN (count(id_disciplina_habilidade) > 0 then true
ELSE false
END
from atividade_disciplina_habilidade where id_atividade = 2091 --idAtividadeEmConfirmação
-- Para o caso de False agregar na msg: É necessário registrar habilidade(s) antes de confirmar.
select CASE
WHEN (count(id) > 0 then true
ELSE false
END
from diario where id_atividade = 2091 --idAtividadeEmConfirmação
-- Para o caso de False agregar na msg: É necessário registrar diário(s) antes de confirmar.

select CASE
WHEN (select count(c.id)
from atividade a
left join horario h on (a.id_horario = h.id)
left join turma t on (h.id_turma = t.id)
left join credito c on (t.id = c.id_turma)
where a.id = 2091 --IdAtividadeSelecionada
and c.id_disciplina = 129) - count(*) > 0 then false
ELSE true
END
from frequencia f
where STRPOS(f.frequencia::text, 1) = 0
and f.id_atividade = 2091 --IdAtividadeSelecionada
-- Para o caso de False agregar na msg: É necessário registrar frequência(s) para todos os alunos antes de confirmar.

2. Caso a verificação retorne uma ou mais textos agregados na msg o sistema deve exibir a mensagem com o(s) texto(s) e não permitir o registro a dataPrevista na dataRealizacao e o registro do idFuncionario do usuariLogado no idFuncionario;

3. Caso a verificação não retorne nenhum dos textos agregados na msg o sistema deve registro a dataPrevista na dataRealizacao e o registro do idFuncionario do usuariLogado no idFuncionario;

RN009
select a.id, a.data_prevista, t.nome as diaSemana, a.data_realizacao,
r.nome as regimeLetivo, sf.sigla as bimestre, p.nome_usual
from atividade a
left join horario h on (a.id_horario = h.id)
left join sub_fase_execucao sfe on (a.id_sub_fase_execucao = sfe.id)
left join sub_fase sf on (sfe.id_sub_fase = sf.id)
left join funcionario f on (a.id_funcionario = f.id)
left join pessoa p on (f.id_pessoa = p.id)
left join tipo_dia_semana t on (h.id_tipo_dia_semana = t.id)
left join tipo_regime_letivo r on (h.id_tipo_regime_letivo = r.id)
where a.id = --IdAtividadeSelecionada

RN010
-- O campo frequência da tabela frequência é um array dinâmico de inteiros
-- Para cada frequência armazenará uma quantidade de inteiros de acordo com o nr de horaAtividades prevista para cada atividade
-- Para cada valor inteiro armazenado no array representará o seguinte estado para frequência(inteiro): P = Presença, AO = Atividade Domiciliar, F=Falta, FJ = Falta Justificada e T = Transferido
-- Enquanto a atividade não tiver frequência registrada o sistema representará estado I para todas as posições do array da frequência
-- Após o primeiro registro de frequência e não for declinado o estado P (Presença), AO (Atividade Domiciliar), F (Falta), FJ(Falta Justificada) e T (transferido) o sistema deverá salvar o estado I (Indefinido)
-- Para cada item do array deverá ser apresentando um icone que represente os quatro estados (I = Indefinido, AO = Atividade Domiciliar, P = Presença, F = Falta, FJ = Falta Justificada e T = Transferido)
-- O sistema deverá salvar o registro de frequência individualmente a cada ação realizada no registro de um aluno

RN011 (Issue #98)
-- Para os alunos que NÃO tenham um tipo de frequência pré-definido na lista (RN007) devem ser ofertadas as seguintes opções:
--- A ação [P] deverá atribuir estado P (Presença) para a posição do array da frequência que foi acionado
--- A ação [F] deverá atribuir estado F (Falta) para a posição do array da frequência que foi acionado
--- A ação [AO] deverá atribuir estado AO (Atividade Domiciliar) para a posição do array da frequência que foi acionado
-- Para os alunos que tenham um tipo de frequência pré-definido na lista (RN007), apenas alunos pré-definido como AO devem ser ofertadas as opções TAD, TF, e os alunos pré-definido como FJ, C ou T ficam bloqueados sem opções de mudança
- A ação [Anular Reposição] deverá estar disponível somente para os registros de frequência em que a "Data de reposição" estiver preenchida
-- Ao acionar a ação [Anular Reposição] o sistema deverá limpar o campo "Data de reposição" da frequência

RN012 (Issue #98)
-- Para os alunos que NÃO tenham um tipo de frequência pré-definido na lista (RN007):
--- A ação [TP] deverá atribuir estado P (Presença) para todas as posições do array de frequência para todos alunos da lista
--- A ação [TF] deverá atribuir estado F (Falta) para todas as posições do array de frequência para todos alunos da lista
--- A ação [Limpar] deverá atribuir estado I (Indefinido) para todas as posições do array de frequência para todos alunos da lista
-- Para os alunos que tenham um tipo de frequência pré-definido na lista (RN007) registrar o tipo pré-definido para cada aluno

RN013 (Issue #98)
-- Para os alunos que NÃO tenham um tipo de frequência pré-definido na lista (RN007):
--- A ação [TP] deverá atribuir estado P (Presença) para todas as posições do array de frequência para todos alunos da lista
--- A ação [TF] deverá atribuir estado F (Falta) para todas as posições do array de frequência para todos alunos da lista
--- A ação [Limpar] deverá atribuir estado I (Indefinido) para todas as posições do array de frequência para todos alunos da lista
-- Para os alunos que tenham um tipo de frequência pré-definido na lista (RN007) registrar o tipo pré-definido para cada aluno

Kronos

https://kronos/gerenciarDiario.html

Gerenciar Diário

Curso RN001 | Periodo RN002 | Turma RN014 | Disciplina RN003 | Bimestre RN015 (Issue #56)

Gerar Atividades | Excluir dias não letivos

RN022 Atividades	Registrar Realização
[Data Prevista] [Data Realização]	[Confirmar] [Editar] [Visualizar]
[Data Prevista] [Data Realização]	[Confirmar] [Editar] [Visualizar]

RN013 (-1) (-1) (-1)

Direitos | Objetivos | Competências | Habilidades

Direitos RN004 (Issue #56)	Acoes
[Código] [BNCC] [Bimestre] [Descrição...]	[Excluir] [Visualizar]
[Código] [BNCC] [Bimestre] [Descrição...]	[Excluir] [Visualizar]

(-3) (-2)

Diário

RN015 (-10)

Tipo Programa | Observação

Procedimento | Recurso

Tipo Programa | Observação

Procedimento | Recurso

RN001
select distinct c.id, c.id_nivel, c.nome
from curso c
left join periodo p on (c.id = p.id_curso)
left join disciplina d on (p.id = d.id_periodo)
left join horario h on (d.id = h.id_disciplina)
left join funcionario f on (h.id_funcionario = f.id)
left join turma t on (h.id_turma = t.id)
where f.encerrada = false
and c.id_instituicao = ((instituicaoUsuarioLogado
and f.id_pessoa = ((pessoaUsuarioLogado

RN002
select distinct pe.id, p.id_faixa_ano, p.nome
from periodo_execucao pe
left join turma t on (pe.id = t.id_periodo_execucao)
left join horario h on (d.id = h.id_turma)
left join funcionario f on (h.id_funcionario = f.id)
left join periodo p on (pe.id_periodo = p.id)
where p.id_curso = 4 = ((CursoSelecionado
and f.id_pessoa = 4 --idPessoaUsuarioLogado

RN003
select distinct d.id, d.sigla
from horario h
left join disciplina d on (h.id_disciplina = d.id)
left join funcionario f on (h.id_funcionario = f.id)
where h.id_turma = 20 -- idTurmaSelecionada
and f.id_pessoa = 4 --idPessoaUsuarioLogado

RN014
select distinct t.id, t.ano, t.sigla
from turma t
left join horario h on (t.id = h.id_turma)
left join funcionario f on (h.id_funcionario = f.id)
where f.id_periodo_execucao = 20 -- idPeriodoExecucaoSelecionado
and f.id_pessoa = 2 --idPessoaUsuarioLogado

RN004 (Issue #56)
select ad.id_atividade, ad.id_disciplina_direito, d.id, d.codigo, d.bncc, sf.sigla, d.descricao
from atividade_disciplina_direito ad
left join disciplina_direito dd on (ad.id_disciplina_direito = dd.id)
left join direito d on (dd.id_direito = d.id)
left join sub_fase sf on (dd.id_sub_fase = sf.id)
where ad.id_atividade = --IdAtividadeSelecionada
order by d.codigo

RN009 (Issue #56)
select d.id, d.codigo, d.bncc, sf.sigla, d.descricao
from disciplina_direito dd
left join direito d on (dd.id_direito = d.id)
left join sub_fase sf on (dd.id_sub_fase = sf.id)
where (dd.data_final_vigencia is null or dd.data_final_vigencia <= current_date)
and dd.id_disciplina = idDisciplinaSelecionada
and not exists
(select ad.id_disciplina_direito
from atividade_disciplina_direito ad
where ad.id_atividade = --IdAtividadeSelecionada
and ad.id_disciplina_direito = dd.id)
and dd.id_sub_fase = --idSubFaseSelecionada ou condição omitida se todos
order by d.codigo

RN005 (Issue #56)
select ao.id_atividade, ao.id_disciplina_objetivo, o.id, o.codigo, o.bncc, sf.sigla, o.descricao
from atividade_disciplina_objetivo ao
left join disciplina_objetivo dob on (ao.id_disciplina_objetivo = dob.id)
left join objetivo o on (dob.id_objetivo = o.id)
left join sub_fase sf on (dob.id_sub_fase = sf.id)
where ao.id_atividade = --IdAtividadeSelecionada
order by o.codigo

RN006 (Issue #56)
select ac.id_atividade, ac.id_disciplina_competencia, c.id, c.codigo,
c.bncc, sf.sigla, c.descricao
from atividade_disciplina_competencia ac
left join disciplina_competencia dc on (ac.id_disciplina_competencia = dc.id)
left join competencia c on (dc.id_competencia = c.id)
left join sub_fase sf on (dc.id_sub_fase = sf.id)
where ac.id_atividade = --IdAtividadeSelecionada
order by c.codigo

Selecionar Direito

Bimestre RN023 (Issue #56)

RN009 (Issue #56) Direitos	Seleção
[Código] [BNCC] [Bimestre] [Descrição...]	[X]
[Código] [BNCC] [Bimestre] [Descrição...]	[ ]

Cancelar (-1) | Selecionar RN010 (-1)

RN010 (Issue #56)
select dob.id, o.codigo, o.bncc, sf.sigla, o.descricao
from disciplina_objetivo dob
left join objetivo o on (dob.id_objetivo = o.id)
left join sub_fase sf on (dob.id_sub_fase = sf.id)
where (dob.data_final_vigencia is null or dob.data_final_vigencia <= current_date)
and dob.id_disciplina = idDisciplinaSelecionada
and not exists
(select ad.id_disciplina_objetivo
from atividade_disciplina_objetivo ad
where ad.id_atividade = --IdAtividadeSelecionada
and ad.id_disciplina_objetivo = dob.id)
and dob.id_sub_fase = --idSubFaseSelecionada ou condição omitida se todos
order by o.codigo

RN008
select c.id, c.codigo, c.bncc, sf.sigla, c.descricao
from disciplina_competencia dc
left join competencia c on (dc.id_competencia = c.id)
left join sub_fase sf on (dc.id_sub_fase = sf.id)
where (dc.data_final_vigencia is null or dc.data_final_vigencia <= current_date)
and dc.id_disciplina = idDisciplinaSelecionada
and not exists
(select ad.id_disciplina_competencia
from atividade_disciplina_competencia ad
where ad.id_atividade = --IdAtividadeSelecionada
and ad.id_disciplina_competencia = dc.id)
and dc.id_sub_fase = --idSubFaseSelecionada ou condição omitida se todos
order by c.codigo

Visualizar Direito

Nível | BNCC | Instituição

Código | Ativo

Direito (400 caracteres)

Campos de experiências

CE Campo de experiência 1

CE Campo de experiência 2

Fechar (-1)

RN012
select d.id, d.procedimento, d.recurso, d.observacao, t.nome as tipoPrograma
from diario d
left join tipo_programa t on (d.id_tipo_programa = t.id)
where d.id_atividade = --IdAtividadeSelecionada

RN013
Ao acionar a ação "Confirmar" o sistema deverá registrar na atividade a dataPrevista na dataRealizacao e registrar o idFuncionario do usuariLogado no idFuncionario

RN016
Ao acionar a ação "Selecionar" o sistema deverá registrar os direitos selecionados na tabela atividade_disciplina_direito vinculando-os a atividade selecionada

RN015
Ao acionar a ação "Selecionar" o sistema deverá registrar os objetivos selecionados na tabela atividade_disciplina_objetivo vinculando-os a atividade selecionada

RN025
Ao acionar a ação "Selecionar" o sistema deverá registrar as competencias selecionadas na tabela atividade_disciplina_competencia vinculando-os a atividade selecionada

RN021
Ao acionar a ação "Selecionar" o sistema deverá registrar as habilidades selecionadas na tabela atividade_disciplina_habilidade vinculando-os a atividade selecionada

RN017
select a.id, a.data_prevista, t.nome as diaSemana, a.data_realizacao,
r.nome as regimeLetivo, sf.sigla as bimestre, p.nome_usual
from atividade a
left join horario h on (a.id_horario = h.id)
left join sub_fase_execucao sfe on (a.id_sub_fase_execucao = sfe.id)
left join sub_fase sf on (sfe.id_sub_fase = sf.id)
left join funcionario f on (a.id_funcionario = f.id)
left join pessoa p on (f.id_pessoa = p.id)
left join tipo_dia_semana t on (h.id_tipo_dia_semana = t.id)
left join tipo_regime_letivo r on (h.id_tipo_regime_letivo = r.id)
where a.id = --IdAtividadeSelecionada

Direitos | Objetivos | Competências | Habilidades

Direitos RN005 (Issue #56)	Acoes
[Código] [BNCC] [Bimestre] [Descrição...]	[Excluir] [Visualizar]
[Código] [BNCC] [Bimestre] [Descrição...]	[Excluir] [Visualizar]

(-5) (-4)

RN011 (Issue #56)
select dh.id, h.codigo, h.bncc, sf.sigla, h.descricao
from disciplina_habilidade dh
left join habilidade h on (dh.id_habilidade = h.id)
left join sub_fase sf on (dh.id_sub_fase = sf.id)
where (dh.data_final_vigencia is null or dh.data_final_vigencia <= current_date)
and dh.id_disciplina = idDisciplinaSelecionada
and not exists
(select ad.id_disciplina_habilidade
from atividade_disciplina_habilidade ad
where ad.id_atividade = --IdAtividadeSelecionada
and ad.id_disciplina_habilidade = dh.id)
and dh.id_sub_fase = --idSubFaseSelecionada ou condição omitida se todos
order by h.codigo

RN007 (Issue #56)
select ad.id_atividade, dh.id, disciplina_habilidade, h.id, h.codigo,
h.bncc, sf.sigla, h.descricao
from atividade_disciplina_habilidade dh
left join disciplina_habilidade dh (dh.id_disciplina_habilidade = dh.id)
left join habilidade h on (dh.id_habilidade = h.id)
left join sub_fase sf on (dh.id_sub_fase = sf.id)
where ad.id_atividade = --IdAtividadeSelecionada
order by h.codigo

Selecionar Objetivo

Bimestre RN023 (Issue #56)

RN010 (Issue #56) Objetivos	Seleção
[Código] [BNCC] [Bimestre] [Descrição...]	[X]
[Código] [BNCC] [Bimestre] [Descrição...]	[ ]

Cancelar (-1) | Selecionar RN012 (-1)

RN016 (Issue #85)
--Gerar atividades
-- 1º Executa a consulta
select h.id, h.data_inicial, h.data_final, h.regular, h.id_funcionario,
h.id_disciplina, h.id_turma, h.id_tipo_dia_semana, h.id_tipo_regime_letivo
from horario h
left join funcionario f on (h.id_funcionario = f.id)
where h.id_turma = 20 --idTurmaSelecionada
and f.id_pessoa = 2 --idPessoaUsuarioLogado
and h.id_disciplina = 47 --idDisciplinaSelecionada
--2º Executar o seguinte algoritmo:
-- Para cada registroCorrente retornado da consulta
-- percorrer da datainicial até a dataFinal do registroCorrente
-- se o diaSemana da dataCorrente for igual ao tipoDiaSemana do registroCorrente
-- verificar se já existe uma atividade registrada com essa data/horário/funcionario
-- e caso não exista
e não conste da lista gerada por RN024
e exista um idSubFaseExecucao Válido =
(
select sfe.id
from horario h
left join turma t on (h.id_turma = t.id)
left join periodo_execucao pe on (t.id_periodo_execucao = pe.id)
left join fase_execucao fe on (pe.id = fe.id_periodo_execucao)
left join sub_fase_execucao sfe on (fe.id = sfe.id_fase_execucao)
where h.id = 2 --idHorarioCorrente
and '30'::Z0' between sfe.data_inicio and sfe.data_fim --dataPrevistaRegistroCorrente
)
-- inserir a atividade com
-- dataPrevista = dataCorrente
-- dataRealizacao = null
-- idHorario = idRegistroCorrente
-- idFuncionario = idFuncionarioRegistroCorrente
-- idSubFaseExecucao

RN022
select a.id, a.data_prevista, a.data_realizacao
from atividade a
left join horario h on (a.id_horario = h.id)
left join funcionario f on (h.id_funcionario = f.id)
where h.id_turma = 20 -- idTurmaSelecionada
and f.id_pessoa = 2 --idPessoaUsuarioLogado
and h.id_disciplina = 53 --idDisciplinaSelecionada
and a.id_sub_fase_execucao = 60--IdSubFaseExecucaoSelecionado

RN023 (Issue #56)
select sf.id, sf.sigla
from sub_fase sf
left join fase f on (sf.id_fase = f.id)
left join periodo p on (f.id_periodo = p.id)
where p.id_curso = 4 = ((CursoSelecionado
and f.id = 53 --idFaseSelecionada
-- A combo de seleção de conter a opção sf.id (Null) e sf.sigla (Todos) + as selecionadas pela query acima
-- Na combo deverá vir pré-selecionada a sub-fase já selecionada na RN015 (sf.id) e a consulta executada pelo filtro
-- Quando selecionado Todos deve ser omitido o filtro de sub-fase nas RN006, RN009, RN010, RN011

RN025 (Issue #51)
delete from atividade where id in (
select a.id
from atividade a
left join horario h on (a.id_horario = h.id)
left join funcionario f on (h.id_funcionario = f.id)
where h.id_turma = 18-- idTurmaSelecionada
and f.id_pessoa = 5903 --idPessoaUsuarioLogado
and h.id_disciplina = 111 --idDisciplinaSelecionada
and exists (
select x.dia_nao_letivo from (
select generate_series (data_hora_inicio::date, data_hora_final::date, '1 day'::interval) as dia_nao_letivo
from evento
where id in (select id from evento where dia_letivo = false and id_calendario = 2) x --idCalendarioCursoDisciplina
where x.dia_nao_letivo = a.data_prevista
));

RN024 (Issue #51)
select generate_series (data_hora_inicio::date, data_hora_final::date, '1 day'::interval) as dia_nao_letivo
from evento
where id in (select id from evento where dia_letivo = false and id_calendario = 2) --idCalendarioCursoDisciplina

Selecionar Competência

Bimestre RN023 (Issue #56)

RN008 (Issue #56) Competências	Seleção
[Código] [BNCC] [Bimestre] [Descrição...]	[X]
[Código] [BNCC] [Bimestre] [Descrição...]	[ ]

Cancelar (-1) | Selecionar RN009 (-1)

Visualizar Competência

Nível | BNCC | Instituição | Geral | Código

Componente | Ativo

Competência (500 caracteres)

Fechar (-1)

Visualizar Habilidade

Nível | BNCC | Instituição | Código | Ativo

Componente

Habilidade (1500 caracteres)

Faixas / Anos

BE Bebês (zero a 1 ano e 6 meses) | CR Crianças bem pequenas (1 ano e 7 meses a 3 anos e 11 meses) | 1A 1º Ano | 2A 2º Ano

Eixo (100 caracteres)

Campo de atuação (300 caracteres)

Prática de linguagem (300 caracteres)

Objeto de conhecimento (500 caracteres)

Unidade temática (300 caracteres)

Competências

Competências vinculadas
[Nível] [BNCC] [Geral] [Código] [Componente...] [Descrição ...] [Ativo]
[Nível] [BNCC] [Geral] [Código] [Componente...] [Descrição ...] [Ativo]

Fechar (-1)

Direitos | Objetivos | Competências | Habilidades

Direitos RN007 (Issue #56)	Acoes
[Código] [BNCC] [Bimestre] [Descrição...]	[Excluir] [Visualizar]
[Código] [BNCC] [Bimestre] [Descrição...]	[Excluir] [Visualizar]

(-6) (-6)

Text

Selecionar Habilidade

Bimestre RN023 (Issue #56)

RN011 (Issue #56) Competências	Seleção
[Código] [BNCC] [Bimestre] [Descrição...]	[X]
[Código] [BNCC] [Bimestre] [Descrição...]	[ ]

Cancelar (-1) | Selecionar RN012 (-1)

Incluir / Alterar Diário

Horas Atividades

HA 07:30 - 08:20 | HA 08:20 - 09:10 | HA 09:10 - 10:00

Data Prevista | Dia Semana | Data Realização | Bimestre | Tipo Programa

Procedimento

Recurso

Observação

Cancelar (-1) | Confirmar (-1)

Alterar Atividade RN017

Horas Atividades

HA 07:30 - 08:20 | HA 08:20 - 09:10 | HA 09:10 - 10:00

Data Prevista | Dia Semana | Regime Letivo

Data Realização | Bimestre | Professor

Cancelar (-1) | Confirmar (-1)

Visualizar Atividade RN017

Horas Atividades

HA 07:30 - 08:20 | HA 08:20 - 09:10 | HA 09:10 - 10:00

Data Prevista | Dia Semana | Regime Letivo

Data Realização | Bimestre | Professor

Fechar (-1)