

1 SISTEMA DESENVOLVIDO

Será desenvolvido um sistema que otimiza a alocação das salas facilitando a vida do gerente. Por se tratar de um problema específico fica difícil encontrar tecnologias disponíveis para a resolução do problema sendo assim necessário o desenvolvimento de um sistema que atenda todas as necessidades exigidas.

1.1 Modelagem

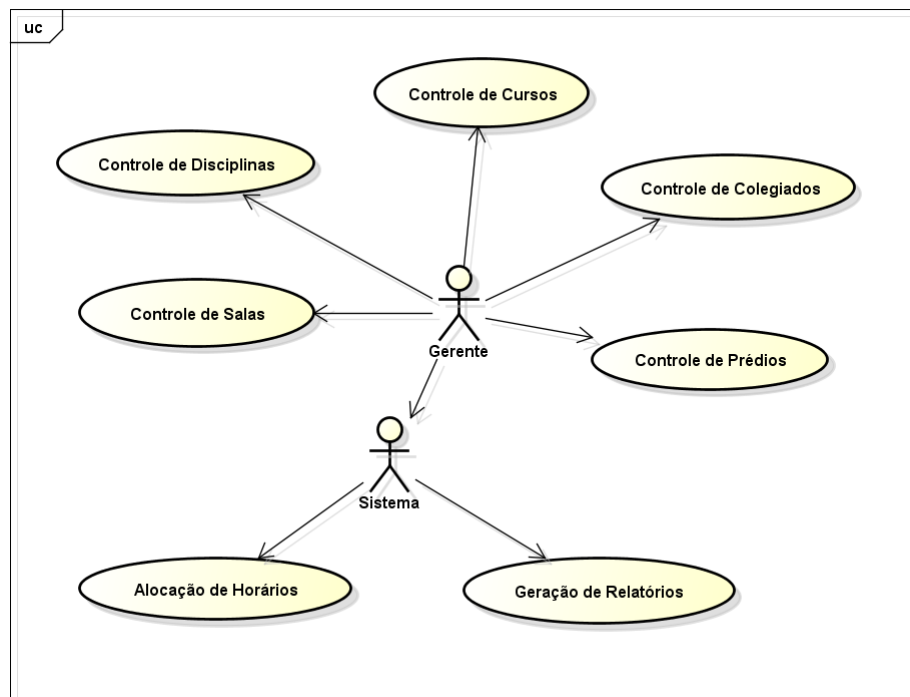
1.1.1 Diagramas de caso de uso

Arrumar referencia

Segundo Booch et al. (2001) a modelagem de um diagrama de caso de uso é uma técnica usada para descrever e definir os requisitos funcionais de um sistema. Para sistemas que possuem um número elevado de funcionalidades, a construção destes diagramas visa facilitar o entendimento do problema, a documentação do que será desenvolvido bem como facilita o próprio desenvolvimento.

A Figura XX descreve todas as funcionalidades que o sistema possui, essas funcionalidades foram divididas em 2 atores "Gerente" e "Sistema" cada um ligado com suas respectivas funcionalidades, porém, o "Gerente" acessa o "Sistema" para ter acessos às funcionalidades do mesmo. O sentido das setas informa o que cada ator pode acessar.

Figura 1 – Diagrama de Caso de Uso



Fonte: Desenvolvido pelo autor

1.1.2 Diagrama de Entidade Relacionamento

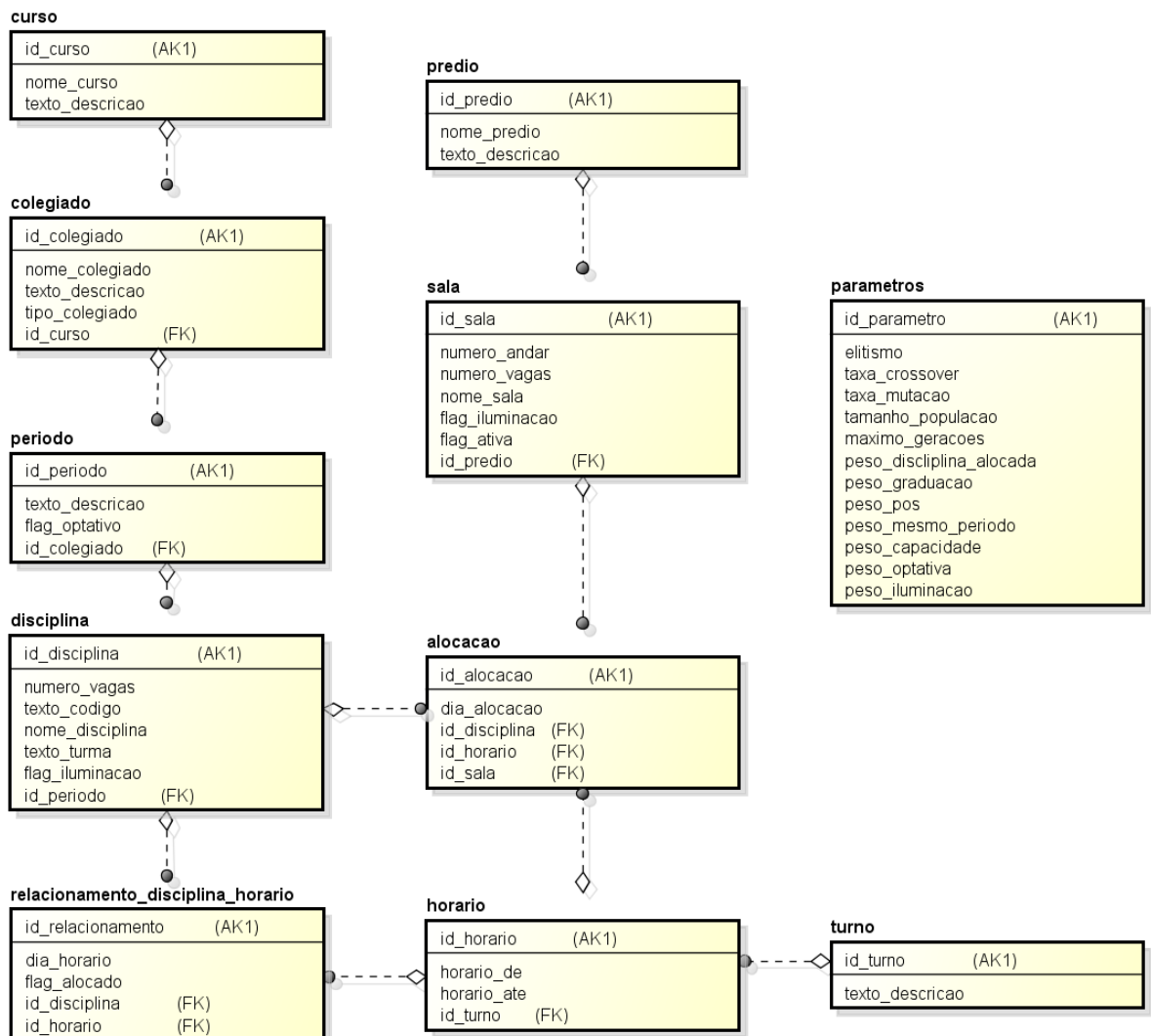
Arrumar referencia

Por se tratar de uma aplicação de banco de dados, a modelagem de dados foi também construída. Segundo Elmasri Navathe (2005) a modelagem conceitual é uma fase muito importante no planejamento de uma aplicação de um banco de dados bem sucedida. Ainda segundo Elmasri Navathe (2005), o modelo relacional representa o banco de dados como uma coleção de relações. Informalmente, cada relação se parece com uma tabela de valores.

Na Figura XXX tem-se o diagrama para o esquema do banco de dados relacional do sistema.

Breve explicação de cada tabela?

Figura 2 – Diagrama de Entidade Relacionamento



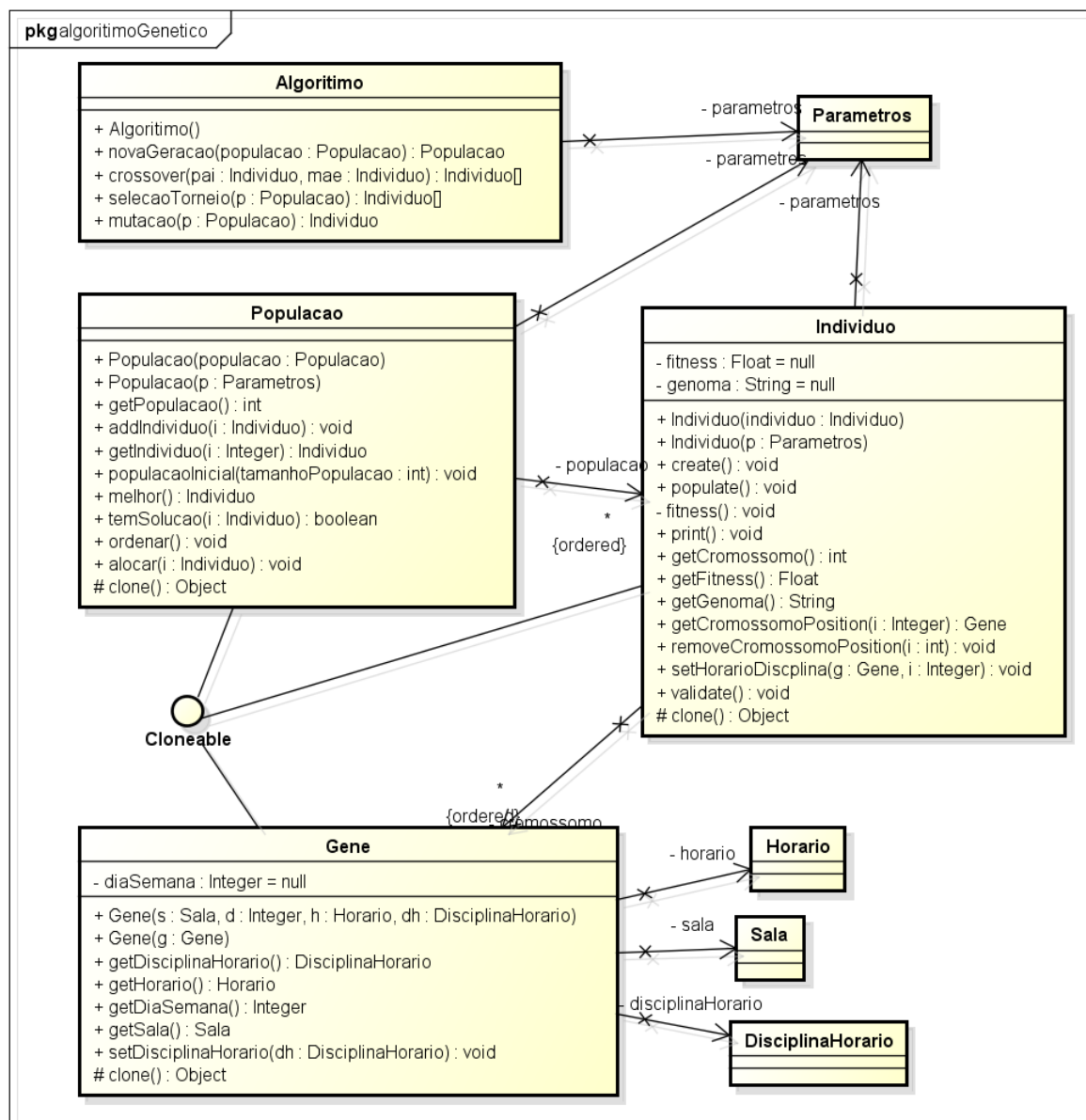
Fonte: Desenvolvido pelo autor

1.1.3 Diagramas de classe

Quando terminar o código revisar.

Segundo Booch et al. (2001), Diagrama de Classes demonstram a estrutura estática das classes de um sistema, onde estas representam as “coisas” que são gerenciadas pela aplicação modelada. Ainda segundo Booch et al. (2001), uma classe num diagrama pode ser diretamente implementada utilizando-se uma linguagem de programação orientada a objetos, no caso deste trabalho, a Linguagem Java. A partir da Figura 3.7 pode-se observar as classes manipuladores do sistema. Elas estão identificadas, descritas com seus métodos e relacionadas entre si. Geralmente um sistema possui mais de um diagrama, pois nem todas se encaixam em um diagrama específico.

Figura 3 – Diagrama de Classe



Fonte: Desenvolvido pelo autor

1.2 Algoritmo Genético

O primeiro passo a ser dado para o desenvolvimento do algoritmo é o pleno conhecimento do problema e a ligação entre os termos utilizados na biologia com os itens do problema proposto anteriormente. Foram utilizadas varias fontes para o desenvolvimento do trabalho, porém foram feitas modificação para que o modelo tratado por outros autores funcionasse como necessario.

Para o melhor entendimento dos passos tomados durante a interpretação do problema e da sua ligação com o algoritmo genetico será criado um pequeno ambiente de alocação. Este ambiente contem as seguintes propriedades, três salas, seis horarios, sete dias da semana, cinco disciplinas e nove relacionamentos de obrigatoriedade entre disciplina e horário.

Figura 4 – Representação do ambiente

Horário			
Id	Turno	Horário de	Horário até
1	Manhã	07:30	09:30
2	Manhã	09:30	12:00
3	Tarde	13:00	15:30
4	Tarde	15:30	18:00
5	Noite	19:00	20:30
6	Noite	20:30	22:30

Relacionamento Disciplina Horário		
Id	Horário	Disciplina
1	1	1
2	2	1
3	1	2
4	1	2
5	3	3
6	4	3
7	3	4
8	4	4
9	6	5

Disciplina			
Id	Período	Nome	Vagas
1	1	História do Brasil	90
2	2	História Medieval	60
3	3	História Antiga	60
4	3	História da Africa	60
5	4	História Moderna	30

Sala			
Id	Nome	Número de vagas	Iluminacao
1	1011	90	Clara
2	1012	60	Clara
3	1013	30	Escura

Fonte: Desenvolvido pelo autor

1.2.1 Indivíduo

Alguns trabalhos tratam cromossomos e individuos pela mesma representação biológica, neste caso cromosso neste trabalho o termo cromossomo se refere a sequencia de genes e o termo Indiviuo será a combinação de cromossomo e fitness.

O termo Gene representa uma combinação de quatro variaveis Sala, Dia da Semana, Horário, e o relacionamento entre Disciplina Horário. As três primeiras variaveis são fixas e não podem ser nulas pois o conjunto de Genes forma um cromossomo que é a alocação de todas as disciplinas em horarios diferentes.

Um Gene com o relacionamento Disciplina Horário igual a nulo, representa um horario vago, como exemplo podemos descrever a seguinte situação, mostra que nos determinados parametros não existe nenhuma sala alocada.

Figura 5 – Representação Gene

Gene			
Sala	Dia da Semana	Horário	Relacionamento
2	3	4	2

Gene			
Sala	Dia da Semana	Horário	Relacionamento
2	3	5	null

Fonte: Desenvolvido pelo autor

Um cromossomo é uma sequência de genes o que representa uma alocação completa que engloba todas as salas, todos os dias da semana, e todos os horários disponíveis para alocação de disciplinas. Uma vez que este valor não é variável temos um cromossomo com um valor fixo, que serão inseridos os horários disponíveis para alocação das disciplinas. O tamanho do cromossomo é medido pela seguinte fórmula (Número de Salas * Número de Horários * Número de dias da semana) neste ambiente o valor é igual a 210 Genes que compõe o cromossomo. A representação binária do cromossomo se deve ao item relacionamento do Gene estar preenchido ou não 1 para preenchido e zero para nulo. A representação gráfica do cromossomo apenas id do relacionamento para uma melhor visualização.

Figura 6 – Representação Cromossomo

Cromossomo							
1	null	null	3	4	null	...	7

Fonte: Desenvolvido pelo autor

Por fim indivíduo é a combinação do cromossomo e a pontuação adquirida após a execução do método de cálculo de fitness.

Figura 7 – Representação Indivíduo

Indivíduo								
Fitness	Cromossomo							
12.0	1	null	null	3	4	null	...	7

Fonte: Desenvolvido pelo autor

1.2.2 População

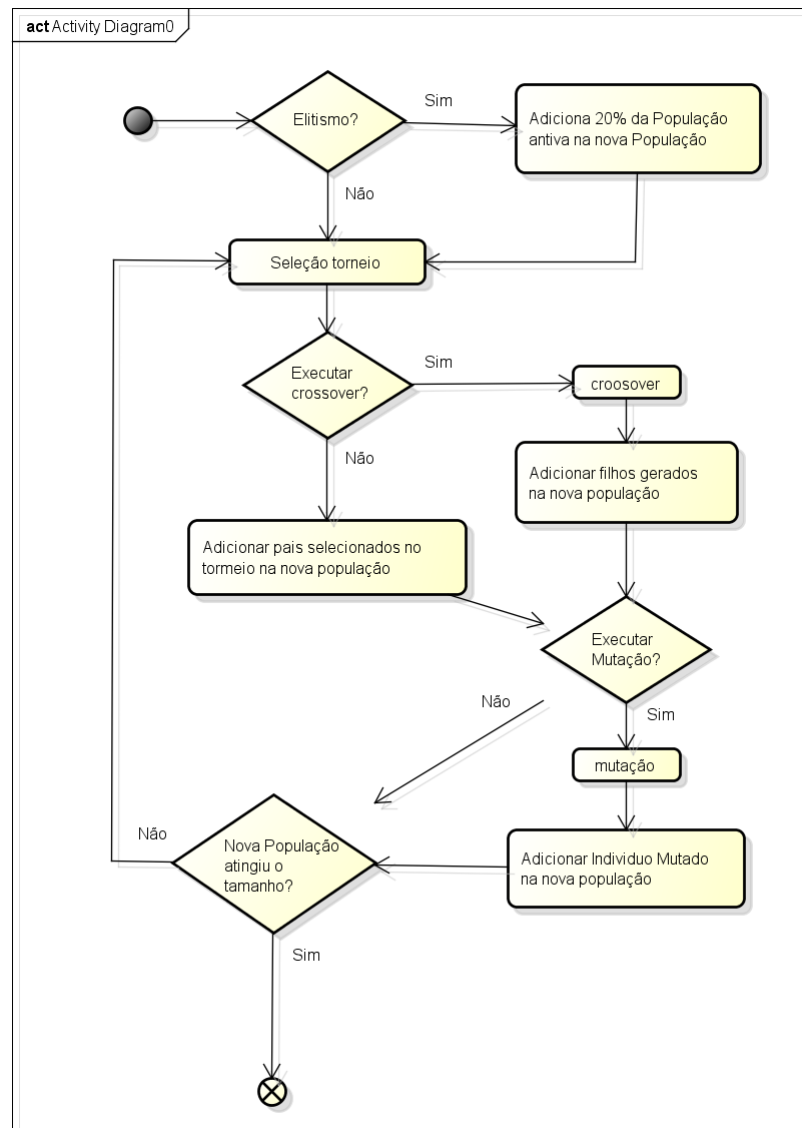
População é o conjunto de Indivíduos, a população tem um número fixo de Indivíduos, e uma porcentagem para que os operadores genéticos possam ser executados. Os Melhores Indivíduos de uma População são indicados pelas maiores pontuações de fitness de cada Indivíduo.

População Inicial cria Indivíduos randômicos até que o número da população máxima seja atingida. Para a criação de cada indivíduo são utilizados operadores randômicos para adicionar o relacionamento DisciplinaHorário em cada um dos Genes que estão previamente criados.

como Nulo. Uma vez que o Indivíduo inicial é populado a quantidade de Genes com relacionamento entre as tabelas DisciplinaHorario não se modificaram para ser manter uma integridade dos dados que serão utilizados para o resultado final do algoritmo.

Nova População é criada a partir de uma População criada anteriormente, se o operador genético elitismo estiver como valor verdadeiro, iniciamos esta nova população com 20% dos melhores Indivíduos da População anterior. Para cada interação do algoritmo até se completar o número total de Indivíduos para aquela nova População, podemos ter duas operações ocorrendo, Mutação e Crossover. Para que estas operações genéticas aconteçam são utilizados valores randômicos para serem comparados com os valores dos parâmetros utilizados para Mutação e Crossover. Em cada interação são selecionados por torneio os pais para o crossover, se der verdadeira a condição para se executar o crossover os pais descartados e os filhos gerados a partir dos Gens dos pais serão adicionados na nova População. Caso a condição de Mutação seja verdadeira um dos itens selecionados no torneio sofre a mutação genética, O Indivíduo antes da mutação genética é descartado e o Indivíduo Mutado é adicionado na nova população.

Figura 8 – Fluxo Nova População



Fonte: Desenvolvido pelo autor

Melhor Indivíduo é aquele que contém o resultado o que tiver a melhor pontuação ao se passar o número de gerações pré definidas.

1.2.3 Operadores Genéticos

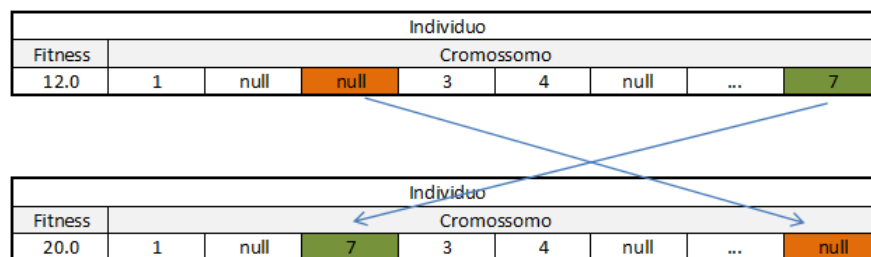
Este trabalho utiliza elitismo como operador genético ao se iniciar uma nova população, vinte por cento dos melhores indivíduos são escolhidos para compor a nova população. O elitismo é calculado através da função objetiva criada para o problema específico do trabalho.

Para selecionar os indivíduos para realizar o crossover é utilizado método de seleção por torneio, são escolhidos três indivíduos da população anterior, e destes três são escolhidos os dois com maior pontuação de fitness, os dois indivíduos selecionados são enviados para o crossover.

Mutação é a inversão genética dos genes de um indivíduo escolhido aleatoriamente da população anterior, após a realização da mutação genética o indivíduo é inserido na nova população.

Primeiramente são escolhidos dois genes do cromossomo, após a escolha aleatória dos itens a serem trocados, é feita a troca dos genes e retornado um indivíduo que tem a composição genética após a alteração. Após a mutação este indivíduo recebe uma nova nota de fitness de acordo com a sua nova sequência de genes e sua adaptação no ambiente, esta nota pode ser maior ou menor do que a anterior.

Figura 9 – Representação Mutação



Fonte: Desenvolvido pelo autor

Crossover é o cruzamento dos indivíduos selecionados pela seleção torneio. Explicar com figuras. após o cruzamento dos indivíduos os dois filhos gerados são inseridos na nova população.

1.2.4 Definição da função objetivo

Somatório disso

Para o cálculo do fitness foram definidos pesos para modelagem da função, estes pesos podem ser configurados de acordo com a necessidade da alocação.

Graduação alocada ganha 05 de peso

Pós graduação alocada ganha 03 de peso

Períodos na mesma sala cada um ganha 05 de peso * o número do período

Quantidade de vagas igual a da sala 05 de peso

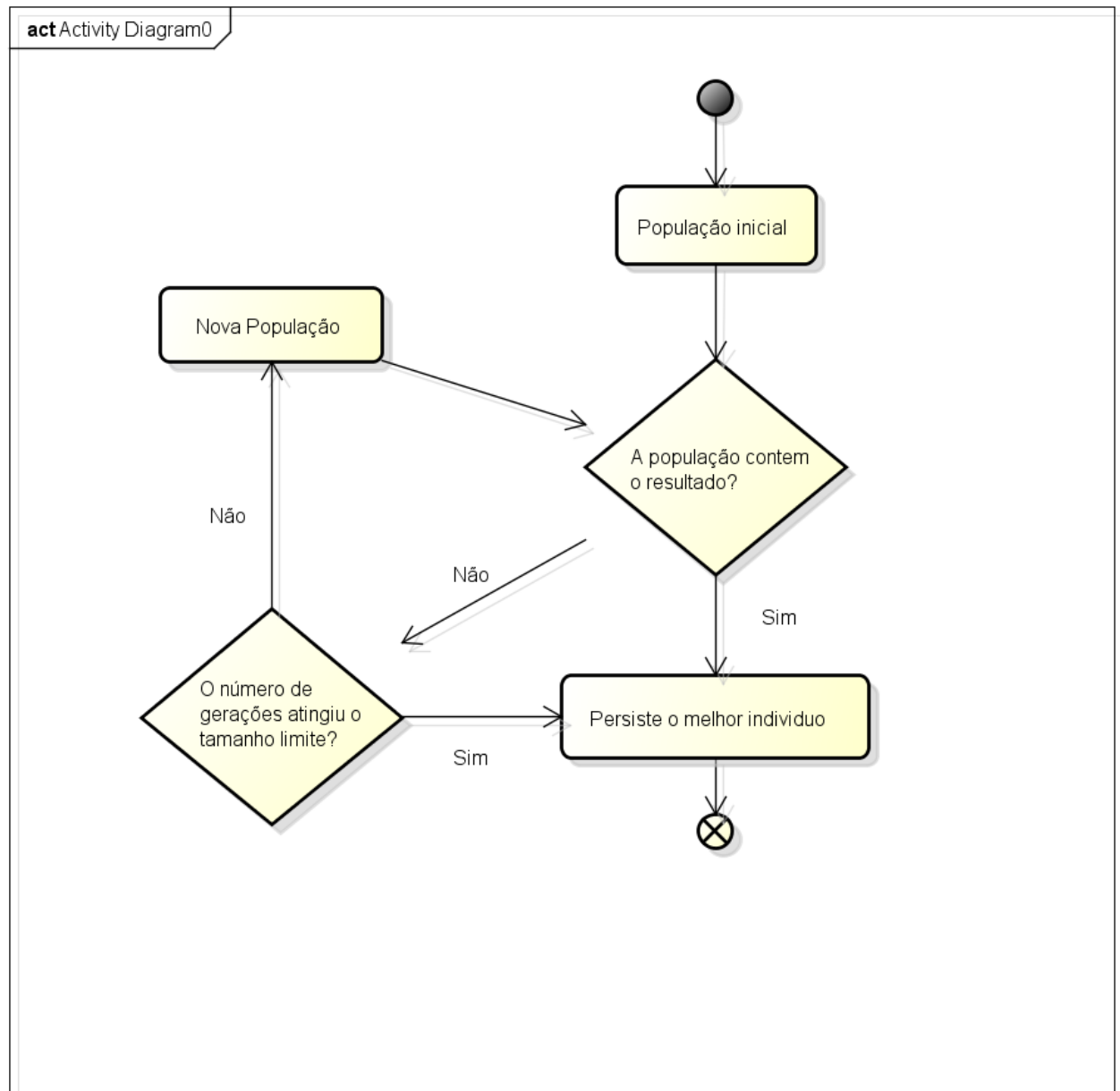
não optativa ganha 5

optativa ganha 3
 eliminacao atendida 5
 Criar a função matemática com as legendas conforme o trabalho 117.pdf
 Falar o numero de salas, o numero de horarios, o numero de cursos o numero de colegia-
 dos o numero de periodos o numero de disciplinas para cada colegiado.....
 restrições
 falar um pouco das restrições e enumeralas
 As disciplinas não podem ser alocadas em horarios diferentes dos que já foram pré
 definidos pelo colegiado.
 As disciplinas devem ter apenas a quantidade de alocações necessarias.
 As disciplinas devem respeitar a capacidade da sala.
 As disciplinas não optativas tem preferencia de alocação na mesma sala.
 Preferencias por salas claras ou escuras
 Restrição 1
 Fitness
 Para se iniciar o calculo do fitness são verificados todos os horarios já alocados somando
 os pesos se adequados.
 para cada gene
 se tem horario alocado
 horario bate
 capacidade da turma
 turma graduacao
 optativa
 eliminacao
 soma tudo
 fim se tem alocação
 soma tudo
 fim para cada gene
 Calculo do fitness01 $\text{somatatorio} \times 100 / \text{colocar algum valor para dividir}$ não sei ainda
 calculo fitness02 penaliza disciplinas com mais alocação do que se deve
 para cada gene
 para cada disciplina
 soma
 fim
 Calculo fitness02 $= \text{fitness01} \times (1 - (\text{total alocados} - \text{total necessario} / \text{dividir pelo numero}$
 possivel de alocações)) fim
 calculo fitness03 penalidade por capacidade
 para cada gente
 se a sala tiver capacidade diferente
 fim
 calculo fitness03 $= \text{fitness02} \times (1 - (\text{numero de erros} / \text{numero de possiveis alocações}))$
 calculo fitness04 preferencias clara ou escura
 para cada gene
 se tiver com o optativo errado
 fim
 calculo fitness04 $= \text{fitness03} \times (1 - (\text{numero de erros} / \text{numero de possiveis alocações}))$
 O fitness04 é o resultado final

1.2.5 Fluxo do algoritmo

O fluxo do algoritmo conforme a imagem XX é iniciado pela criação da população inicial, para cada interação do algoritmo é verificado se a população contém o resultado e se o algoritmo não atingiu o número de gerações pré definidas. Se as duas condições forem falsas o algoritmo cria uma nova População de acordo com a figura XXX

Figura 10 – Fluxo Algoritmo



Fonte: Desenvolvido pelo autor

1.2.6 Parametros Algoritmo

1.3 Arquitetura e Telas do sistema