

# SisFundacoes: Uma Ferramenta para Estimativa da Capacidade de Carga de Fundações Profundas

Gabriel Bronzatti Moro, Eduardo Carvalho da Costa, João Pablo Silva da Silva e Magnos Baroni

**Resumo**—The need to build new buildings is a demand resulting from population growth, new constructions in places where the soil does not offer favorable conditions is the main reason to choose by deep foundations, which allow you to choose a depth where the soil is adequate to support the load of the construction. The design of a deep foundation involves a lot of parameters and many load capacity calculation routines, this process is tiresome for the Civil Engineer and generates problems such as inconsistencies in the results, error in the registered parameters, numeric conversion errors among others. Thus the aim of this paper is to present the SisFundacoes, a tool that offers supports to create projects of load capacity the stakes in deep foundations. The methodology used to develop the tool consists of an incremental development process, which uses basic software engineering disciplines such as problem analysis, design, coding and testing. The main result obtained in this work is a software tool, this tool can be utilized in deep foundations project for to calculate the loading capacity of the used stakes in building. The software tests performed show satisfactory results. from these we can say that the tool has a satisfactory level of reliability and can be used to calculate the load capacity of deep foundations.

**Index Terms**—Software Engineering, Civil Engineering, Load Capacity, Support Tool

## I. INTRODUÇÃO

FUNDAÇÕES profundas são elementos estruturais pré-moldados ou concretados *in situ*, que possuem a função de transmitir ao solo as cargas provenientes de uma supraestrutura. Para obras verticais de maior porte, como edifícios, estádios, centros esportivos, e para regiões onde o solo superficial não apresenta a devida capacidade de carga, opta-se pela execução de fundações profundas para transmitir ao solo as cargas provenientes destas construções [1].

Historicamente a previsão da capacidade de carga destas fundações constitui-se em um dos principais desafios do Engenheiro Geotécnico, pois requer além da aplicação das metodologias de projeto, a estimativa dos parâmetros do solo e da interação solo-estaca. Velloso e Lopes (2010) salientam que “o ramo das fundações exige muita experiência por parte do engenheiro civil” devido à diversidade de solos existentes e a sua imprevisibilidade comportamental.

Gabriel Bronzatti Moro é graduado no curso de Engenharia de Software pela Universidade Federal do Pampa (UNIPAMPA), Campus Alegrete. (e-mail: gabrielbronzattimoro.es@gmail.com)

Eduardo Carvalho da Costa é graduado no curso de Engenharia Civil pela Universidade Federal do Pampa (UNIPAMPA), Campus Alegrete. (e-mail: eduardo.cdc@hotmail.com)

João Pablo Silva da Silva é doutorando em Computação pela Universidade Federal do Rio Grande do Sul (UFRGS) e docente na Universidade Federal do Pampa (UNIPAMPA), Campus Alegrete. (email: joaosilva@unipampa.edu.br)

Magnos Baroni é mestre em Geotecnia pela Universidade Federal do Rio de Janeiro (UFRJ) e docente na Universidade Federal de Santa Maria (UFSM). (email: magnos.baroni@gmail.com)

A prática brasileira utiliza o ensaio SPT (*Standard Penetration Test*) para a obtenção dos parâmetros do solo. Esses parâmetros são posteriormente utilizados em equações analíticas, resultando na estimativa da capacidade de carga de diferentes tipos de estacas, com diferentes diâmetros e comprimentos.

Desta maneira, apresentar-se-á a versão beta da ferramenta SisFundacoes para apoiar o cálculo da capacidade de carga de diferentes tipos de estacas, com diferentes profundidades e inseridas em solos com diferentes composições.

### A. Organização do Trabalho

O presente trabalho está organizado da seguinte maneira: a seção 2 apresenta os conceitos básicos sobre Fundações Profundas, já na seção 3 é apresentado os métodos de cálculo utilizados pelo software, a metodologia utilizada para o desenvolvimento da ferramenta é apresentada na seção 4 e na seção 5 é apresentado o resultado do trabalho. O artigo é finalizado com a seção 6, nela é descrito sobre aprendizados, desafios e o principal resultado adquirido na realização do trabalho.

Novembro 12, 2015

## II. FUNDAÇÕES PROFUNDAS

A NBR 6122/2010 [2] define fundação profunda como aquela que possui profundidade de assente maior do que 2 vezes sua menor dimensão em planta e de, no mínimo, 3 metros.

As fundações profundas são executadas em forma de estacas e sua capacidade de suporte pode ser calculada considerando duas parcelas: a parcela de resistência de ponta da estaca; e a parcela e resistência lateral, que é calculada considerando o atrito lateral que ocorre entre a estaca e o solo. Existindo situações onde a estaca poderá ser dimensionada apenas pela sua resistência de ponta ou pelo seu atrito lateral. Deste modo, a capacidade de carga total de uma estaca é dada pela equação 1.

$$Q_t = Q_p + Q_l \quad (1)$$

Onde:

- $Q_t$  = capacidade de carga total da estaca;
- $Q_p$  = resistência de ponta da estaca;
- $Q_l$  = resistência por atrito lateral da estaca.

O software em desenvolvimento trabalha com a soma das parcelas de resistência, mas também mostra ao usuário as resistências de ponta e por atrito lateral calculadas separadamente, ficando a critério do usuário a sua utilização de melhor maneira.

A. Tipos de Estacas

Conforme Velloso e Lopes [3], pode-se classificar as estacas em dois grandes grupos: as estacas “de deslocamento” e as estacas “de substituição”.

As estacas de substituição são estacas moldadas *in loco*, perante a realização de uma perfuração no solo, através de escavação, onde ocorre a retirada do solo e, logo após, a concretagem da estaca. Nessa categoria enquadram-se as estacas Strauss, rotativas em geral, estacas hélice contínua e estacas raiz.

As estacas de deslocamento são aquelas fabricadas em indústrias e transportadas até a obra. Podem ser de concreto armado, metálicas ou de madeira (no caso de obras provisórias) e são introduzidas no solo através de percussão ou prensagem.

B. Ensaio Standard Penetration Test (SPT)

O ensaio SPT consiste em um ensaio de reconhecimento de solo, realizado em campo, com a finalidade de conhecer o subsolo *in situ* através da retirada de amostras, por meio da cravação de um amostrador e medida pelo número de golpes necessários para a sua cravação.

O ensaio SPT é normatizado pela ABNT/NBR 6484/2001. Nesse ensaio, determina-se o índice de resistência à penetração do amostrador ( $N_{SPT}$ ), além do nível da água no subsolo e da estratificação do solo, por meio da retirada de amostras. O ensaio consiste basicamente na cravação de um amostrador normalizado, chamado originalmente de *Raymond-Terzaghi*, por meio de golpes de um peso de 65 kgf, caindo de uma altura de 75 cm. Conta-se o número de golpes necessários para penetrar cada parcela de 15cm do amostrador. O índice de resistência à penetração ( $N_{SPT}$ ) será a soma de números de golpes necessários para cravar os últimos 30cm do amostrador no solo.

A empresa responsável pela execução do SPT, deve gerar um relatório contendo as principais informações do ensaio, como o tipo de solo, valor do  $N_{SPT}$  a cada metro de profundidade, cota da boca do furo e a localização dos ensaios. Esse relatório serve como base para os cálculos da capacidade de carga das estacas e para o projeto de fundações como um todo.

III. MÉTODOS DE CÁLCULO

Os métodos Aoki e Velloso (1975) e Décourt e Quaresma (1978) são amplamente utilizados na previsão da capacidade de carga de estacas. Esses métodos foram implementados no software e serão detalhados a seguir.

A. Método de Aoki-Velloso

O método é baseado em resultados de resistência do ensaio de Cone (CPT), foi apresentado por Aoki e Velloso em 1975 no V Congresso Panamericano de Mecânica dos Solos e Engenharia de Fundações [5].

A resistência total pode ser definida como a soma das parcelas de resistência lateral e resistência de ponta. A resistência de ponta calculada pelo método de Aoki-Velloso é dada pela equação 2.

$$R_p = \sum K * N_b * (A_b / F_1) \tag{2}$$

Onde:

- K = coeficiente de correlação com os resultados de ensaio de cone;
- $N_b$  = valor de  $N_{SPT}$  na base (ponta) da estaca;
- $A_b$  = área da base (ponta) da estaca;
- $F_1$  = coeficiente de correlação de resistência de ponta para levar em conta a diferença de comportamento entre a estaca e o ensaio de cone.

A resistência lateral calculada pelo método de Aoki-Velloso é calculada pela seguinte fórmula definida na equação 3.

$$R_l = \sum \alpha_i * k_i * N_m * P * (\Delta_L / F_2) \tag{3}$$

Onde:

- $\alpha_i$  = razão de atrito na camada i;
- $N_m$  = valor de  $N_{SPT}$  médio na camada i;
- $\Delta_L$  = comprimento da estaca na camada i;
- $k_i$  = coeficiente de correlação com resultados do ensaio de cone na camada i;
- $F_2$  = coeficiente de correlação de resistência lateral para levar em conta a diferença de comportamento entre a estaca e o ensaio de cone;
- P = perímetro do fuste da estaca.

A Tabela I mostra os valores dos coeficientes k e  $\alpha_i$ .

Tabela I  
VALORES DOS COEFICIENTES K E  $\alpha_i$  [5]

Tipo de Solo	K (MPa)	$\alpha_i$ (%)
Areia	1,0	1,4
Areia siltosa	0,8	2,0
Areia silto-argilosa	0,7	2,4
Areia-argilosa	0,6	3,0
Areia argilosa-siltosa	0,5	2,8
Silte	0,4	3,0
Silte arenoso	0,55	2,2
Silte areno-argiloso	0,45	2,8
Silte argiloso	0,23	3,4
Silte argilo-arenoso	0,25	3,0
Argila	0,2	6,0
Argila arenosa	0,35	2,4
Argila areno-siltosa	0,3	2,8
Argila siltosa	0,2	4,0
Argila silto-arenosa	0,3	3,0

A Tabela II mostra os valores dos coeficientes  $F_1$  e  $F_2$  segundo o tipo de solo.

B. Método Décourt-Quaresma

Foi desenvolvido originalmente para estacas de deslocamento (cravadas), onde os valores  $Q_p$  e  $Q_l$  são obtidos em função dos valores de  $N_{SPT}$  medidos no ensaio de SPT.

A resistência de ponta calculada pelo método de Décourt-Quaresma é dada pela equação 4 [5].

$$Q_p = \alpha * C * N_{p,medio} * A_p \tag{4}$$

Tabela II  
VALORES DOS COEFICIENTES  $F_1$  E  $F_2$  [5]

Tipo de Estaca	$F_1$	$F_2$
Franki de fuste apilado	2,3	3,0
Franki de fuste vibrado	2,3	3,2
Metálica	1,75	3,5
Pré-moldada de concreto	2,5	3,5
Cravada à percussão		
Pré-moldada de concreto crava à prensagem	1,2	2,3
Escavada com lama betonítica	3,5	4,5
Raíz	2,2	2,4
Strauss	4,2	3,9
Hélice contínua	3,0	3,8

Onde:

- $\alpha$  = coeficiente de ponderação;
- $N_{p,medio}$  = média dos valores de  $N_{SPT}$  medidos na ponta de estaca e imediatamente acima e abaixo da ponta da estaca;
- $A_p$  = área da base (ponta) da estaca;
- $C$  = coeficiente em função do tipo de solo.

A resistência lateral calculada pelo método Décourt-Quaresma, em  $kN$  é dada pela equação 5 [5].

$$Q_l = 10 * \beta * (N_l/3 + 1) * U * L \quad (5)$$

Onde:

- $\beta$  = coeficiente de ponderação;
- $N_l$  = valor do  $N_{SPT}$  na profundidade considerada;
- $U$  = perímetro da seção transversal do fuste;
- $L$  = comprimento da estaca.

Nas Tabelas III e IV mostra-se os valores de  $C$ ,  $\alpha$  e  $\beta$  para o método Décourt-Quaresma.

Tabela III  
VALOR DE  $C$  PARA O MÉTODO DÉCOURT-QUARESMA [5]

Tipo de Solo	$C$ ( $kN/m^2$ )
Argila	12
Silte argiloso	20
Silte arenoso	25
Areia	40

Tabela IV  
VALORES DE  $\alpha$  E  $\beta$  PARA O MÉTODO DÉCOURT-QUARESMA [5]

Tipo de Solo $\alpha - \beta$	Argilas	Siltes	Areias
Tipo de Estacas/ $\alpha - \beta$			
Escavada sem lama	0,85 - 0,8	0,6 - 0,65	0,5 - 0,5
Escavada com lama	0,85 - 0,9	0,6 - 0,75	0,5 - 0,6
Hélice contínua	0,3 - 1	0,3 - 1	0,3 - 1
Raíz	0,85 - 1,5	0,6 - 1,5	0,5 - 1,5
Microestaca	1 - 3	1 - 3	1 - 3

#### IV. MATERIAIS E MÉTODOS

A Engenharia de Software é uma área responsável por tratar as preocupações relacionadas ao desenvolvimento de software, tais como a criação de novos sistemas e também a manutenção de softwares existentes. Essa área provê métodos, teorias e ferramentas para que o desenvolvedor selecione-as adequadamente para a resolução de problemas [8].

A metodologia utilizada nesse trabalho compreende em um processo de desenvolvimento incremental, nesse modelo de processo a cada ciclo um incremento é gerado. A seguir será apresentado as disciplinas básicas de Engenharia de Software utilizadas nos ciclos do processo.

##### A. Análise do Problema

O levantamento de requisitos é fundamental para a construção de um software, pois os requisitos não são inventados e sim descobertos através reuniões, entrevistas, técnicas de etnografia, *workshops*, entre outras, as quais são realizadas juntamente com o usuário e demais *stakeholders*<sup>1</sup> [6].

As atividades de análise foram realizadas no decorrer do processo, essas atividades tem por objetivo o conhecimento do problema para o qual a ferramenta será desenvolvida. Dentre as atividades de análise, as principais realizadas foram questionários, entrevistas e *workshops* para compreender o problema enfrentado no projeto de fundações profundas. Foram envolvidos como usuários do produto nessas atividades um professor e um aluno do curso de Engenharia Civil.

Os artefatos gerados durante as atividades de análise foram uma documentação de casos de uso, protótipos de tela, um modelo de domínio e o diagrama de atividades de usuário. A partir do estudo dos domínio foi possível criar a documentação de casos de uso da ferramenta, ela define quais são os requisitos funcionais e não funcionais, o fluxo básico de operações que o usuário deve realizar, como também fluxos de exceção e alternativos. Na Figura 1 é apresentado resumidamente os casos de uso da ferramenta em diagrama.

Segundo Larman [12], um modelo de domínio permite a compreensão dos conceitos associados ao domínio do problema investigado nas atividades de concepção em um processo de software. O modelo de domínio construído nas atividades de análise foi fundamental para compreender a relação dos conceitos envolvidos no contexto de Engenharia Civil, para projetos de estimativa da capacidade de carga de diferentes tipos de fundações profundas. Esse modelo sofreu várias modificações ao longo do processo. Na Figura 2 é possível visualizá-lo.

O diagrama de atividades foi utilizado para mapear as possíveis ações que o usuário pode realizar no sistema. Esse artefato foi construído juntamente com o usuário para também estipular a ordem das tarefas a serem realizadas no sistema. Na Figura 3 é possível visualizar o diagrama de atividades.

##### B. Projeto da Solução

Segundo Pressman [7], as atividades de projeto permitem o conhecimento dos componentes fundamentais do software,

<sup>1</sup>*Stakeholders* é um termo utilizado para representar as partes interessadas em um processo de software [13]



Figura 1. Diagrama de casos de uso.

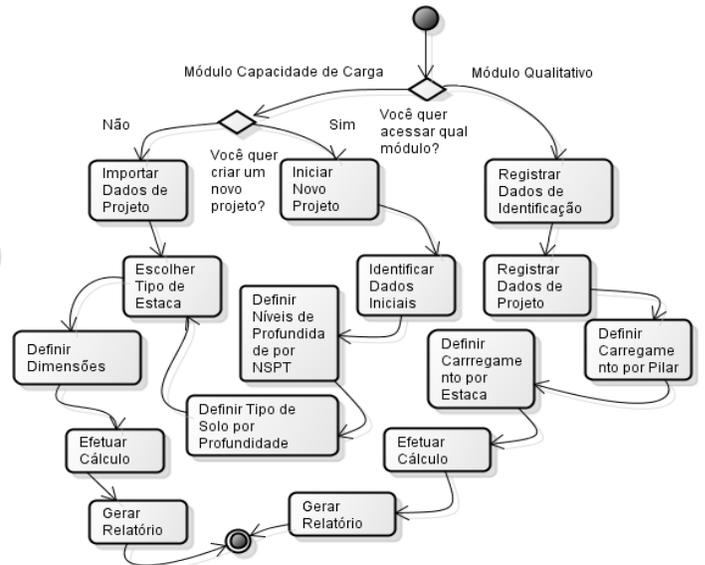


Figura 3. Diagrama de atividades.

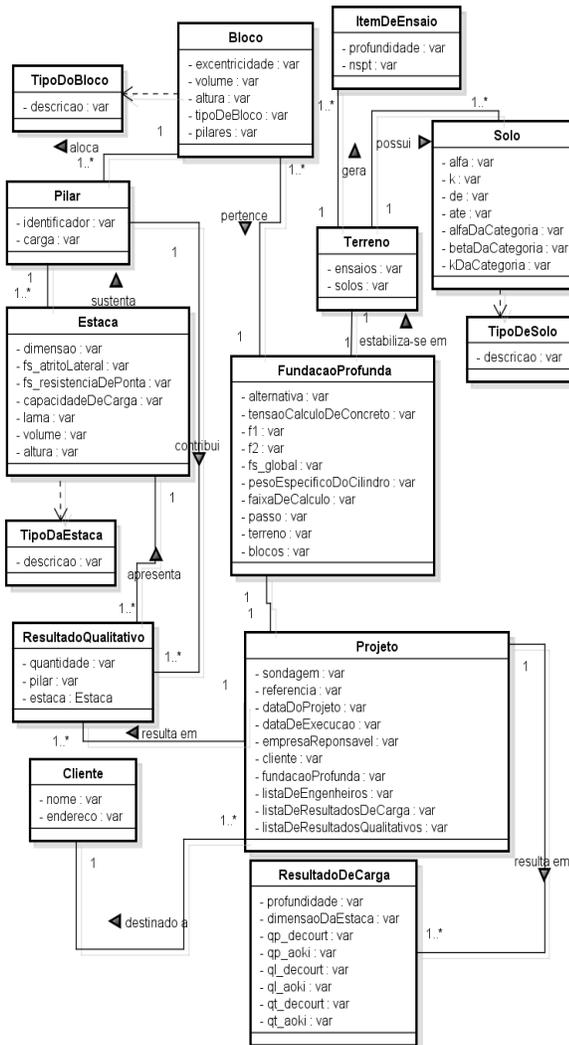


Figura 2. Modelo de domínio gerado nas atividades de análise.

como também suas interfaces de conexão. Os artefatos de pro-

jeito devem permitir que a solução encontrada pelo Engenheiro de Software atenda os requisitos capturados pelas atividades de análise [7].

O principal objetivo das atividades de projeto é a construção da proposta de solução para a problemática investigada nas atividades de análise. Os artefatos de projeto gerados no decorrer do processo foram o diagrama de arquitetura, classes e de sequência. A Figura 4 apresenta o diagrama da arquitetura definida para a ferramenta.

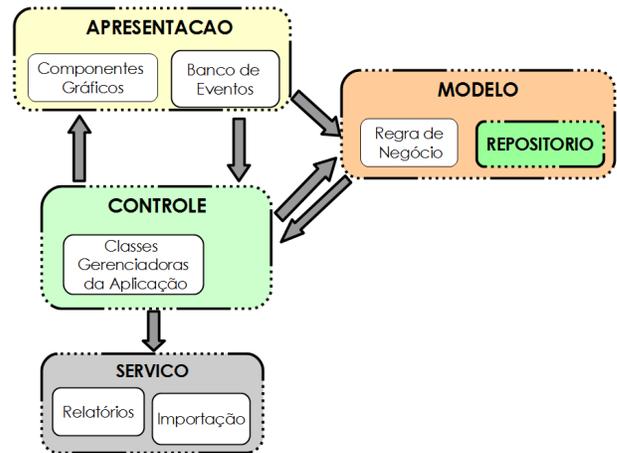


Figura 4. Diagrama da arquitetura da ferramenta.

O formato arquitetural adotado compreende para cada nível da arquitetura uma responsabilidade. O nível apresentação é destinado para tratar questões relacionadas a interface gráfica, ou seja, painéis, botões, formulários, janelas e demais elementos gráficos [10]. O nível modelo trata questões relacionadas a regra de negócio tratada pelo software, a lógica e o processamento das informações. Quanto ao nível serviço, nele são definidos serviços de geração de relatórios, importação de relatórios e demais serviços que são gerenciados pelo

nível de controle e fornecidos para toda a aplicação [10]. No nível controle é realizado o gerenciamento dos demais níveis da arquitetura, essa parte da arquitetura é responsável por gerenciar os serviços, os eventos disparados pelo nível de apresentação e a regra de negócio (localizado no nível modelo) [10]. Também, existe o nível arquitetural repositório, o qual está dentro do nível modelo, esse nível é responsável pelo gerenciamento das coleções de objetos do modelo. Como o software não possui persistência de dados, esse nível é fundamental para o gerenciamento das coleções de objetos em tempo de execução.

### C. Implementação de Código-fonte

A codificação foi realizada utilizando a linguagem de programação Java, o NetBeans foi utilizado como ferramenta de ambiente de desenvolvimento integrado, o controle de versionamento adotado foi o Git com a ferramenta Source-Tree destinada ao gerenciamento do repositório remoto e a documentação foi construída utilizando a API (*Application Programming Interface*) JavaDoc.

A linguagem Java foi escolhida para o desenvolvimento da ferramenta pela vasta quantidade de bibliotecas de programação disponíveis, materiais didáticos e o fato de ser multiplataforma, possibilitando a utilização do mesmo executável para diferentes plataformas de sistemas operacionais (mesma máquina virtual) [11].

A codificação dos métodos foi realizada juntamente com a implementação dos testes unitários, ou seja, o método era codificado e após isso eram construídos os possíveis testes unitários. A partir disso, analisava-se continuamente o histórico de implementação do método, pois a qualquer alteração em seu funcionamento os testes unitários eram novamente executados para avaliar se o método atende ao seu objetivo após a modificação realizada.

### D. Testes de Software

Diferentemente dos testes de caixa branca, os quais levam em consideração o código-fonte do software, o teste caixa-preta leva em consideração apenas as entradas e saídas das funcionalidades do software. Através dele é possível garantir que os componentes do software funcionam isoladamente [9].

A atividade de teste em um processo de desenvolvimento é fundamental, pois garante que os resultados adquiridos durante o desenvolvimento estão em conformidade com os requisitos levantados (testes de usuário); desempenho, segurança e comportamentos previstos em situações extremas (testes de sistema); o funcionamento das partes isoladas (testes unitários); e o funcionamento das partes integradas (testes de integração).

Os testes unitários e de integração foram criados e executados utilizando o *framework* JUnit (provido pelo NetBeans), a cobertura desses testes foi avaliada através do *plugin* Ecmmma.

Através dos testes unitários criados foi possível avaliar como os componentes de software se comportam quando são utilizadas no cálculo diferentes estacas com mesmas dimensões e parâmetros de solo. Dessa maneira, foi possível avaliar o comportamento do software quando executado nessas circunstâncias.

## V. RESULTADOS

### A. Ferramenta de Apoio ao Projeto de Fundações Profundas

O principal resultado adquirido no presente trabalho foi a ferramenta SisFundacoes, destinada a estimativa da capacidade de carga de fundações profundas. Através do mecanismo de navegabilidade entre telas, a ferramenta permite que o usuário realize a inserção de informações e parâmetros para o cálculo da capacidade de carga. Na Figura 5 é apresentado a primeira etapa do procedimento de cálculo.

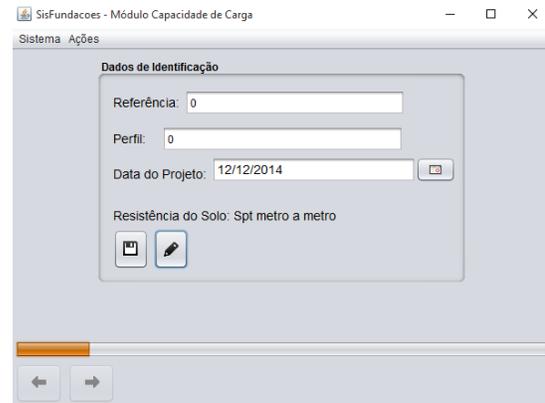


Figura 5. Tela da primeira etapa de projeto.

A primeira etapa do procedimento (UC01 e UC17 da Figura 1) é o cadastro das informações básicas do projeto, como referência, perfil e a data do projeto. Após isso, o usuário define os  $N_{SPTS}$  respectivos de cada profundidade do solo (UC02 e UC03 da Figura 1). A Figura 6 apresenta a tela de profundidade por  $N_{SPT}$ .

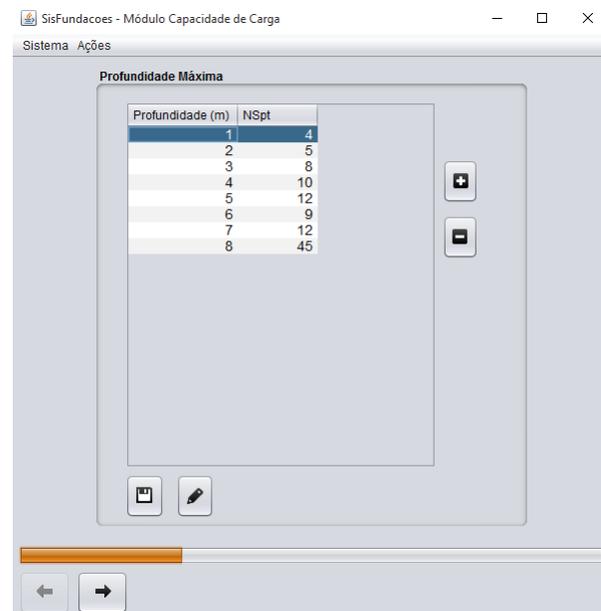


Figura 6. Tela da segunda etapa de projeto.

Após a definição dos  $N_{SPTS}$  por profundidade, o usuário deve informar qual é o tipo de solo por profundidade (UC02

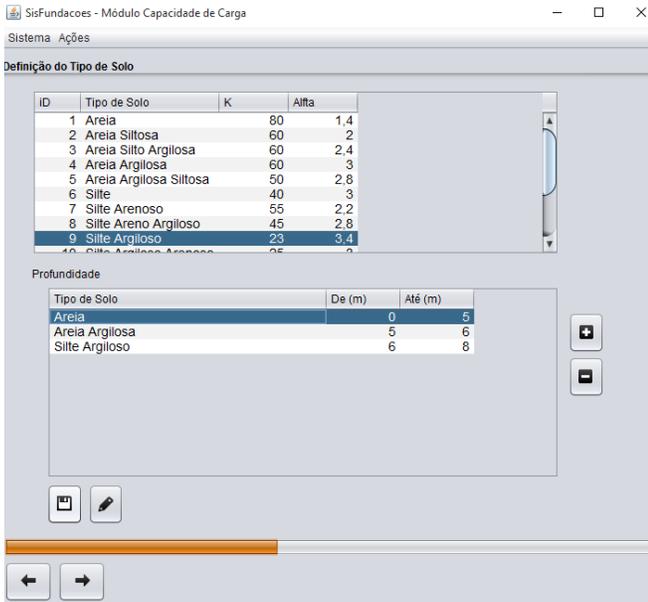


Figura 7. Tela da terceira etapa de projeto.

e UC03 da Figura 1). A Figura 7 apresenta a tela responsável para definir os tipos de solo.

Depois da definição dos tipos de solo por profundidade, o usuário deve escolher qual estaca utilizar e definir as suas dimensões (UC4 e UC5 da Figura 1). Essa etapa do projeto pode ser visualizada na Figura 8.

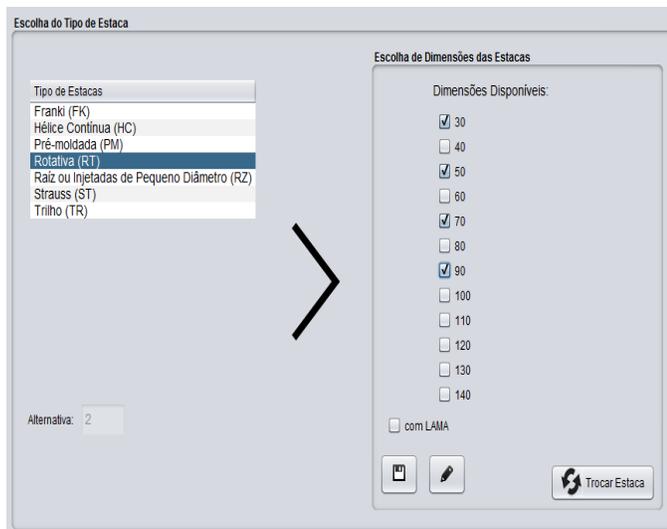


Figura 8. Tela da terceira etapa de projeto.

No quarto passo é definido os fatores de segurança (por padrão eles já vem definidos) e da faixa de cálculo (UC 16 da Figura 1). A faixa de cálculo consiste em estabelecer qual é o início e fim da profundidade que o usuário deseja realizar o cálculo de capacidade de carga.

O último passo do projeto é gerar o relatório de projeto (UC 14 da Figura 1), a ferramenta disponibiliza dois formatos de relatórios: pdf e Excel. O formato Excel permite que posteriormente o usuário importe o relatório para realizar o

cálculo com mesmos parâmetros.

### B. Testes de Caixa-preta

Os tipos de estacas utilizados para para a criação dos testes unitários foram: Trilho, Strauss, Rotativa, Raíz, Pré-moldada, Hélice-contínua e Franki. Para cada estaca foi simulado a mesma condição de terreno, como tipo de solo, profundidade e  $N_{SPT}$  por camada. Como gabarito dos testes unitários foi utilizado a planilha de cálculo criada por Ferreira e Delalibera [15].

O total de casos de teste escritos foram 840, dentre esses, 120 para cada tipo de estaca, dos quais foram divididos em 60 testes para verificar os resultados de capacidade de carga do método de Aoki-Velloso e outros 60 para o método de Décourt-Quaresma. O resultado de cada teste é considerado aprovado quando a capacidade de carga da estaca (resistência total, de ponta e lateral) de ambos os métodos (Aoki-Velloso e Décourt-Quaresma) de cálculo corresponde ao valor definido no gabarito utilizado. Na Figura 9 abaixo é possível visualizar o resultado dos testes em porcentagem.

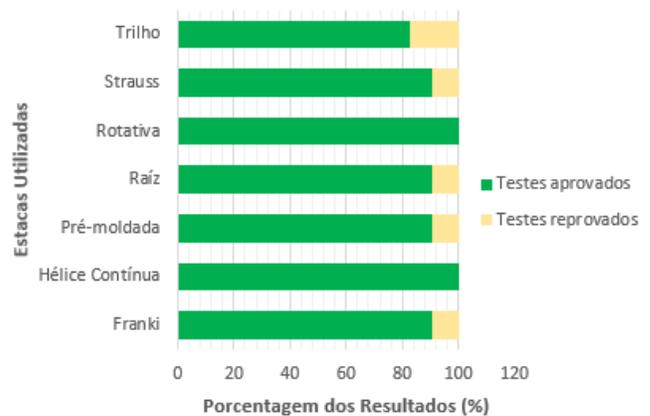


Figura 9. Resultados dos testes unitários.

As estacas onde se obteve 100% de aprovação nos testes foram as estacas de tipo Hélice Contínua e Rotativa, já as estacas de tipo Pré-moldada, Raíz, Strauss e Franki foram obtidos aproximadamente 90,83% de aprovação nos testes. O menor resultado foi obtido com a estaca Trilho, cerca de 82,5%.

O motivo dos resultados não atingirem 100% estão associados aos métodos de arredondamento adotados, os resultados que não foram exatos na porcentagem ficaram na margem de uma unidade acima ou abaixo. Essa margem de resultados é aceitável, a partir dela se pode garantir que a ferramenta é eficiente para projetos de capacidade de carga.

A técnica de análise de cobertura foi utilizada para verificar a porcentagem de estruturas de código exercitadas pelos testes unitários. A Tabela V apresenta as classes alvo e sua porcentagem de cobertura.

A partir dos resultados adquiridos na Tabela V é possível verificar que as classes analisadas possuem cobertura acima de 50%. A classe que obteve menor cobertura foi a classe “Estaca”, esse resultado é justificado, visto que, a mesma

Tabela V  
ANÁLISE DE COBERTURA DE CLASSES ALVO

Nome da Classe	Cobertura
Projeto	64%
EngenheiroCivil	76%
Estaca	61%

possui muitos métodos de acesso (*getters*) e configuração (*setters*) de atributos privados. Quanto a classe “Projeto”, a porcentagem de cobertura indica um índice satisfatório, pois a mesma possui métodos de manipulação para a geração de relatórios de projeto, esses métodos não foram exercitados pelos testes unitários, porque o escopo de teste definido é apenas o cálculo de capacidade de carga das estacas. O melhor índice de cobertura foi o da classe “EngenheiroCivil”, nessa classe estão todas as rotinas de cálculo utilizadas pelo software, logo é a mais exercitada pelos testes.

## VI. CONSIDERAÇÕES FINAIS

Durante o processo de desenvolvimento da ferramenta foi possível aplicar boas práticas de Engenharia de Software, dentre elas: a aplicação de técnicas para a investigação de requisitos, como entrevistas, questionários e *workshops*; padrões arquiteturais e de projetos, como o padrão repositório, *Model-View-Controller (MVC)* e *Factory*; documentação de classes e métodos, utilizando um padrão definido para permitir o entendimento do código; e técnicas de teste, como a utilização da técnica de particionamento por equivalência para definir os testes de caixa-preta.

O objetivo do presente trabalho é apresentar a ferramenta SisFundacoes, a qual oferece suporte para a criação de projetos de dimensionamento de estacas para fundações profundas. A ferramenta permite ao Engenheiro Civil realizar o cálculo de capacidade de carga utilizando os dois métodos de cálculo da estimativa da capacidade, o proposto por Aoki-Velloso e o proposto por Décourt-Quaresma.

A partir dos resultados obtidos nos testes unitários pode-se afirmar que a ferramenta possui um nível satisfatório de confiabilidade quando comparado ao gabarito [15] utilizado como referência nos testes. Nesse sentido, a ferramenta pode ser utilizada tanto por profissionais de Engenharia Civil, como também por alunos em âmbito acadêmico.

### A. Trabalhos Futuros

O próximo passo desse trabalho será aperfeiçoar a ferramenta para que seja possível obter exatidão nos resultados de aceitação dos testes unitários. Também, pretende-se realizar um experimento em ambiente controlado com alunos do curso de Engenharia Civil para coletar informações a respeito dos atributos de usabilidade da ferramenta, os quais influenciam diretamente em sua qualidade.

Além disso, será construído outro módulo na ferramenta, o qual será destinado ao cálculo quantitativo de obras. Esse módulo tem por objetivo informar ao Engenheiro Civil qual é a melhor opção (custo e benefício) de estacas para o seu projeto de fundação profunda. A Figura 10 apresenta os casos de uso do novo módulo.

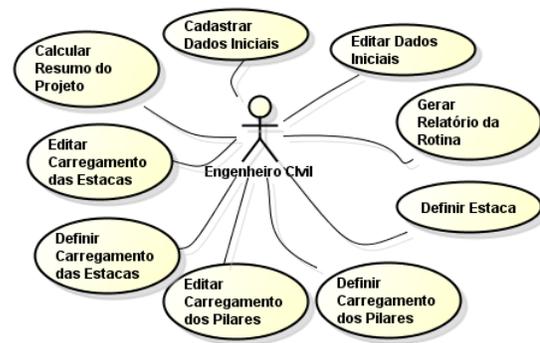


Figura 10. Diagrama de casos de uso do novo módulo.

## REFERÊNCIAS

- [1] Azeredo, H. A., *O Edifício até a sua cobertura*. Edgard Blucher. 2ª ed. São Paulo, 1997.
- [2] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, *NBR 6122: Projeto e Execução de Fundações*. Rio de Janeiro, 2010.
- [3] Velloso. Dirceu e Lopes; Francisco de Rezende, *Fundações: critérios de projeto, investigação do subsolo, fundações superficiais, fundações profundas*. São Paulo, SP: Oficina de Textos, 2010.
- [4] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, *NBR 6484: Solo – Sondagem de simples reconhecimento com SPT – Método de ensaio*. Rio de Janeiro, 2010.
- [5] Décourt. L. Albiero; J.H. e Cintra, *Análise e projeto de fundações profundas*. In: Hachichi, W. et al (Edit.). *Fundações: teoria e prática*. 2ª ed. PINI. p. 265-301, 1998
- [6] Wazlawick. Raul Sidnei, *Análise e Projeto de Sistemas de Informações Orientados a Objetos*. Elsevier. 2ª ed. Rio de Janeiro, 2010.
- [7] Pressmann. Roger S., *Software Engineering: A Practitioner's Approach*. AMGH. 6ª ed. Mc Graw Hill, 2005.
- [8] Sommerville. Ian, *Engenharia de Software*. Pearson. 8ª ed. São Paulo, 2007.
- [9] Wazlawick. Raul Sidnei, *Engenharia de Software: Conceitos e Práticas*. Campus. 1ª ed. São Paulo, 2013.
- [10] Gama. Erich., *Padrões de Projetos: Soluções Reutilizáveis*. Bookman. São Paulo, 2009.
- [11] Deitel. H. M.; Deitel. P. J., *Java, como programar*. Pearson Prentice Hall. 6ª ed. tradução: Edson Furmankiewicz. São Paulo, 2009.
- [12] Larman. Craig, *Utilizando UML e Padrões*. Bookman. São Paulo, 2002.
- [13] Hurt. Robert L., *Sistemas de Informações Contábeis: Conceitos Básicos e Temas Atuais*. AMGH. 3ª ed. São Paulo, 2004.
- [14] Aoki. N; Velloso. D., *An Approximate method to estimate the bearing capacity of piles*. In: PANAMERICAN CONFERENCE ON SOIL MECHANICS AND FOUNDATIONS ENGINEERING, 5., 1975. Buenos Aires. Proceedings... Buenos Aires, 1975.
- [15] Ferreira, T. R; Delalibera, R. G., *Capacidade de Carga em Fundações por Estacas*. In: <http://pt.scribd.com/doc/136503412/Planilha-fundacoes.xls>. Universidade Federal de Goiás. Catalão, 2006.