



FUNVIC
FUNDAÇÃO UNIVERSITÁRIA VIDA CRISTÃ

FACULDADE DE PINDAMONHANGABA



Organização
das Nações Unidas
para a Educação,
a Ciência e a Cultura

Fundação Universitária
Vida Cristã - Brasil
Membro do Movimento de Clubes,
Centros e Associações para a UNESCO

Rodrigo Reis Alves

PROPOSTA DE ARQUITETURA DE SOFTWARE PARA DRONES QUADROTORES

**Pindamonhangaba – SP
2016**



FUNDAÇÃO UNIVERSITÁRIA VIDA CRISTÁ
FUNVIC

FACULDADE DE PINDAMONHANGABA



Fundação Universitária
Vida Cristá - Brasil
Membro do Movimento de Clubes,
Centros e Associações para a UNESCO

Organização
das Nações Unidas
para a Educação,
a Ciência e a Cultura

Rodrigo Reis Alves

PROPOSTA DE ARQUITETURA DE SOFTWARE PARA DRONES QUADROTORES

Artigo apresentado como parte dos requisitos para obtenção do diploma graduação pelo curso de Sistemas de Informação da Faculdade de Pindamonhangaba.

Orientador: Prof. Rodrigo Ramos de Oliveira

**Pindamonhangaba – SP
2016**

Alves, Rodrigo Reis;

Proposta De Arquitetura De Software Para Drones Quadrotores / Rodrigo Reis Alves / Pindamonhangaba-SP : FUNVIC Faculdade de Pindamonhangaba, 2016.

*f. : il.

Artigo (Bacharelado em Sistemas de Informação) FUNVIC-SP.

Orientador: Prof. Rodrigo Ramos Oliveira.

1 Arquitetura de software. 2 Drone. 3 Arduino. 4 Quadricóptero.

I Proposta De Arquitetura De Software Para Drones Quadrotores. II Rodrigo Reis Alves.

Rodrigo Reis Alves
Protótipo de luva com sensores ultrassônicos

Artigo apresentado como parte dos requisitos para obtenção do diploma graduação pelo curso de Sistemas de Informação da Faculdade de Pindamonhangaba.

Data: _____

Resultado: _____

BANCA EXAMINADORA

Prof . _____ Faculdade de Pindamonhangaba

Assinatura _____

Prof . _____ Faculdade de Pindamonhangaba

Assinatura _____

Prof . _____ Faculdade de Pindamonhangaba

Assinatura _____

Este trabalho foi escrito em forma de artigo científico; sob as normas da Revista Eletrônica de Ciências Humanas da FUNVIC; e as normas de publicação encontra-se em anexo.

**PROPOSTA DE ARQUITETURA DE SOFTWARE PARA DRONES
QUADROTORES**
A SOFTWARE ARCHITECTURE PROPOSAL FOR QUADROTOR DRONES

Rodrigo Reis Alves¹ – FUNVIC – Faculdade de Pindamonhangaba
Rodrigo Ramos de Oliveira² – FUNVIC – Faculdade de Pindamonhangaba*
rodrigohd7@gmail.com

Resumo

O Veículo aéreo não tripulado (VANT) ou *drone* (do inglês, zangão) é toda e qualquer aeronave que não necessita de pilotos embarcados para ser guiada. Criada para fins militares, este tipo de aeronave foi projetada e construída para uso em missões de elevado risco humano. A programação de um VANT quadrotor requer funções básicas responsáveis por: i) realizar leituras de posição e movimento do corpo, ii) realizar controle de estabilidade angular nos eixos espaciais x, y e z, iii) realizar controle de estabilidade linear nos eixos espaciais x, y e z, iv) realizar comunicação com dispositivos externos, v) interpretar e executar comandos externos. Neste contexto este trabalho apresenta uma proposta de design de uma arquitetura lógica capaz de realizar os controles operacionais de *drones* quadrotores, com a meta de permitir a implementação de camadas especializadas, escalabilidade de recursos e fácil manutenção do software. O objetivo de propor e avaliar uma arquitetura de programação elaborada para operar um drone quadrotor foi atingido neste trabalho. Os resultados positivos atingidos nos testes práticos realizados com o protótipo sugerem que a arquitetura proposta permite a implementação de camadas especializadas, escalabilidade de recursos e fácil manutenção do software.

Palavras chave: Arquitetura de software. Drone. Arduino. Quadricóptero.

Abstract

Unmanned aerial vehicles (UAV) or drones, is any type of aircraft that does not require any pilots to be guided. Built for military purposes, this type of aircraft has been designed and built for use in missions of high human risk. Programming a UAV Quadrotor requires basic functions responsible for: i) to perform position and attitude readings, ii) to perform angular stability control on the spatial axes x, y and z, iii) to perform linear stability control on the spatial axes x, y and z, iv) to communicate with external devices, v) to interpret and execute external commands. In this context the present work shows a proposal of design of a logical architecture capable of performing the operational controls of drones. With the goal of enabling the implementation of expert layers, scalability of resources and easy maintenance of the software. The objective of proposing and evaluating an elaborate programming architecture to operate a quadrotor drone was reached in this work. The positive results achieved in the practical tests performed with the prototype suggest that the proposed architecture allows the implementation of expert layers, scalability of resources and easy maintenance of the software.

Keywords: Software architecture. Drone. Arduino. Quadcopter.

1 Introdução

O Veículo aéreo não tripulado (VANT) ou *drone* (do inglês, zangão) é toda e qualquer aeronave que não necessita de pilotos embarcados para ser guiada. Criada para fins militares, este tipo de aeronave foi projetada e construída para uso em missões de elevado risco humano. Um *drone* de múltiplas pás rotativas é uma das mais complexas máquinas voadoras, considerando sua habilidade de decolar e aterrissar verticalmente, realizar um voo pairado, executar manobras de arfagem e rolagem mantendo sua altitude, além de deter seu movimento abruptamente (BENICÍ, 2016).

Vários trabalhos acadêmicos foram realizados com o objetivo de se obter um modelo matemático que descreva a dinâmica de voo de um *drone* quadrotor (TAHIR et al, 2016; CHAMBERLIN, 2016; GOODMAN et al, 2015; KAYA, 2015; POUNDS, 2010), como também o controle e estabilização do mesmo (ALEXIS,2012; PAIVA, 2015; SÁ, 2013; LENGGERKE, 2013).

A programação de um VANT quadrotor requer funções básicas responsáveis por: i) realizar leituras de posição e movimento do corpo, ii) realizar controle de estabilidade angular nos eixos espaciais x, y e z , iii) realizar controle de estabilidade linear nos eixos espaciais x, y e z, iv) realizar comunicação com dispositivos externos, v) interpretar e executar comandos externos

Neste contexto o presente trabalho apresenta uma proposta de design de uma arquitetura lógica capaz de realizar os controles operacionais de *drones* quadrotores, com a meta de permitir a implementação de camadas especialistas, escalabilidade de recursos e fácil manutenção do software. A validação de cada recurso contido no design da arquitetura proposta foi obtida por meio de testes práticos executados em protótipo dedicado a este fim.

2 Objetivo

Este trabalho tem como meta propor uma arquitetura lógica baseada na arquitetura de software em camadas que permita a implementação de camadas especialistas, escalabilidade de recursos e fácil manutenção do software.

3 Método

O presente trabalho compõe as seguintes etapas:

- Montagem dos componentes físicos e eletrônicos do protótipo de teste. Nesta etapa foi feita a montagem do protótipo sobre a placa Arduino.
- Pesquisa e elaboração da arquitetura de programação do sistema de operação do protótipo.
- Testes para a validação da arquitetura de programação desenvolvida.

4 Modelo Dinâmico

A construção de um software para o controle e estabilização de um quadrotor deve levar em consideração seu comportamento e estrutura física, o que é necessário durante a implementação de suas estruturas de controle.

Um quadrotor pode ser imaginado como uma estrutura rígida em forma de cruz com quatro motores montados equidistantemente do seu centro de gravidade conforme ilustrado na Figura 2.

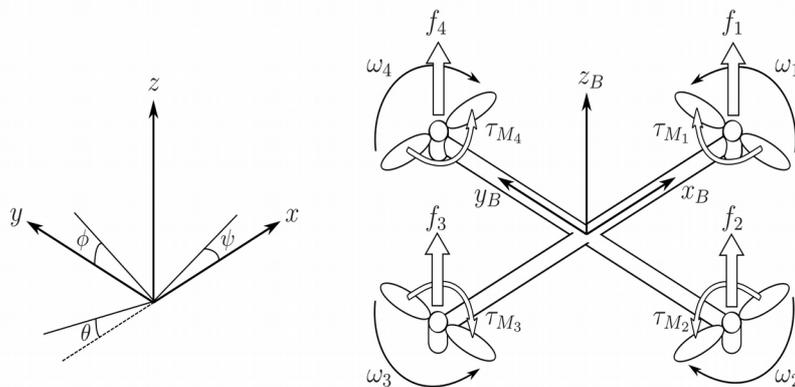


Figura 2 – Modelo dinâmico de um drone quadrotor (ELEMENT 14 COMMUNITY, 2016)

No quadrotor ilustrado na Figura 2 são representadas quatro forças de empuxo (f_1, f_2, f_3, f_4) que são geradas através da velocidade de rotação de cada hélice $(\omega_1, \omega_2, \omega_3, \omega_4)$ fixa nos motores. O empuxo coletivo $(f_c = f_1 + f_2 + f_3 + f_4)$ é a soma das potências geradas por cada

motor e para que haja o equilíbrio de forças o sistema deve respeitar a condição, onde mg é o peso do drone.

As velocidades verticais obtidas pelos motores geram momentos em cada um de seus respectivos braços. Estas quatro forças quando aplicadas originam estados de saída nos componentes $x, y, z, \psi, \theta, \phi$ onde:

- x : posição frontal em relação ao eixo global
- y : posição lateral em relação ao eixo global
- z : posição vertical em relação ao eixo global
- ψ : ângulo *Yaw* (em torno do eixo z)
- θ : ângulo *Pitch* (em torno do eixo x)
- ϕ : ângulo *Roll* (em torno do eixo y)

O ângulo θ (*Pitch*) representa a rotação do corpo em torno do eixo y . O ângulo ϕ (*Roll*) representa a rotação em torno do eixo x . O ângulo ψ representa a rotação em torno do eixo Z .

A potência T em direção ao eixo z é criada pela combinação das forças geradas pelos motores. O torque τ_B é criado pela combinação dos torques τ_ψ , τ_θ e τ_ϕ em direção ao seu correspondente ângulo na posição do corpo do drone. Onde o torque τ_ψ é comumente chamado de movimento de guinada, o torque τ_θ chamado de movimento de arfagem, e o torque τ_ϕ chamado de movimento de rolagem.

A partir do quadrotor ilustrado na Figura 2, o movimento de rolagem é produzido reduzindo a velocidade dos motores 1 e 4 e aumentando a velocidade dos motores 2 e 3 ou vice-versa. O movimento de arfagem é produzido aumentando a velocidade nos motores 1 e 2 e diminuindo a velocidade dos motores 3 e 4 ou vice-versa. O movimento de guinagem é produzindo aumentando a velocidade dos motores 1 e 3 e diminuindo a velocidade dos motores 2 e 4 ou vice-versa.

5 Arquitetura Do Protótipo

A arquitetura do protótipo é ilustrada na Figura 1, seus elementos estão descritos em subseções seguintes.

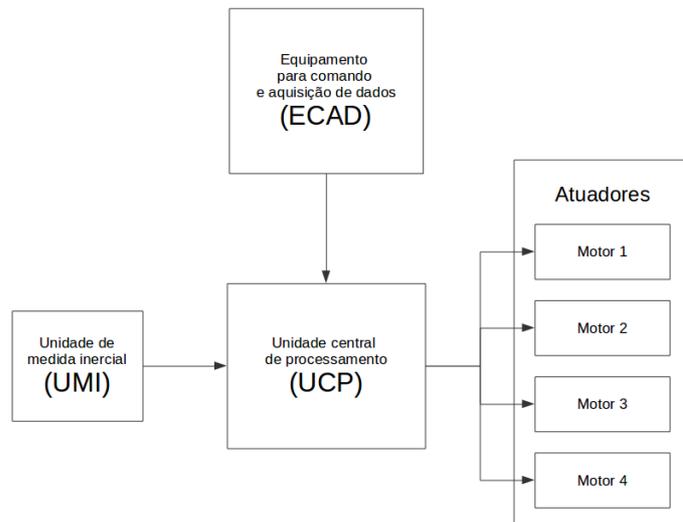


Figura 1 – Arquitetura do protótipo

5.1 Equipamento de comando e aquisição de dados (ECAD)

O ECAD é um equipamento para a realização de operações de leitura e escrita de dados na Unidade central de processamento (UCP), que pode funcionar como um controle remoto permitindo ao usuário acionar comandos disponíveis pelo sistema presente na UCP estabelecendo estados de controle à mesma.

5.2 Unidade de medida inercial (UMI)

Sistemas para determinação de posição normalmente consistem de giroscópios, acelerômetros e magnetômetros, onde, um dispositivo de medição de movimentos contendo acelerômetros e giroscópios é comumente chamado de *Inertial Measurement Unit* (IMU) (FUX, 2008).

Para este trabalho foi utilizado o sensor de movimento MPU6050 que segundo (____, 2016) combina um giroscópio de 3 eixos e um acelerômetro de 3 eixos em um mesmo hardware que processa os algoritmos de fusão de movimentos e fornece leituras sobre os movimentos processados por um protocolo de barramento I²C (MANKAR, 2014). Os pacotes de saída desse sensor representam as acelerações lineares nos eixos x, y e z do plano cartesiano assim como as velocidades angulares dos mesmos.

Para que os sinais recebidos da UMI sejam normalizados foi utilizado um filtro complementar (HIGGINS, 1975), onde foram combinados os sinais de aceleração linear e velocidades angular dos respectivos eixos (x, y, z) para se obter uma boa estimativa de orientação compensando assim o desvio de estado do giroscópio e a dinâmica lenta do acelerômetro.

5.3 Unidade central de processamento (UCP)

A UCP realiza as operações pertinentes a processamentos sensoriais e atuação, onde também possui meios para realizar a comunicação de dados com o ECAD. As funções desse bloco podem ser desempenhadas por um micro controlador, pois esse sistema é constituído de um microprocessador, memória e periféricos de entrada/saída e pode ser programado para funções específicas, como, por exemplo, o controle de máquinas e diferentes automações (CAVALCANTE et all., 2011).

5.4 Bloco de atuadores

O bloco de atuadores compreende atuadores do tipo motores de corrente contínua. Os motores de corrente contínua são relativamente compactos se comparados a outros tipos de motores, e comumente o valor de torque permanece em uma faixa constante para grandes variações de velocidade (BOLDEA e NASAR, 1997).

5.5 Programa de gerenciamento da UCP

O programa de gerenciamento da UCP proposta neste trabalho foi elaborado para atender as funções principais deste protótipo, Calibragem de sensores e Execução de voo.

Durante a calibragem de sensores o sistema deve efetuar as leituras dos mesmos e determinar seus desvios, após deve calcular os valores de *offset* destes sensores, que são valores que deverão compensar os erros medidos. Após calibrados os sensores o sistema deve salvar os valores de *offset*, para que sejam utilizados pelo sistema durante o modo de voo.

Durante o voo o sistema deverá executar leituras da UMI, que fornecerá os sinais pertinentes à postura em referência ao plano inercial. Logo após deverá calcular os sinais de erro e efetuar as

correções de sua postura por meio de controladores PID, considerando também os comandos e sinais provenientes do ECAD (WESCOTT, 2000).

6 Arquitetura Lógica

Conforme a complexidade do software cresce, o *design* do mesmo vai além dos algoritmos e da estrutura de dados. Questões estruturais incluem organização, protocolos de comunicação, acesso e sincronia de dados, e performance (GARLAN e SHAW, 1993). Estes fatores compreendem o problema a ser resolvido em um nível arquitetural e tocam os atributos de qualidade (também conhecidos como requisitos não funcionais) como confiabilidade, disponibilidade, desempenho entre outros.

Estilos arquiteturais definem a família de um dado sistema em termos de padrões estruturais de organização. Mais especificamente, um estilo arquitetural determina o vocabulário dos componentes e conectores que são utilizados nas instâncias do estilo, bem como sua organização referentes a estes componentes.

Um sistema com arquitetura em camadas é hierarquicamente organizado, cada camada é formada por um conjunto de classes com um determinado propósito, demanda serviços das camadas abaixo e fornece serviço para as camadas acima. Em alguns sistemas as camadas internas são escondidas de todas as outras camadas a não ser as camadas adjacentes a mesma, exceto por algumas funções que são cuidadosamente selecionadas para exportação (SHAW e GARLAN, 1996).

Uma ilustração da arquitetura em camadas aplicada a este trabalho é apresentada na Figura 4, seus elementos estão descritos em subseções seguintes.

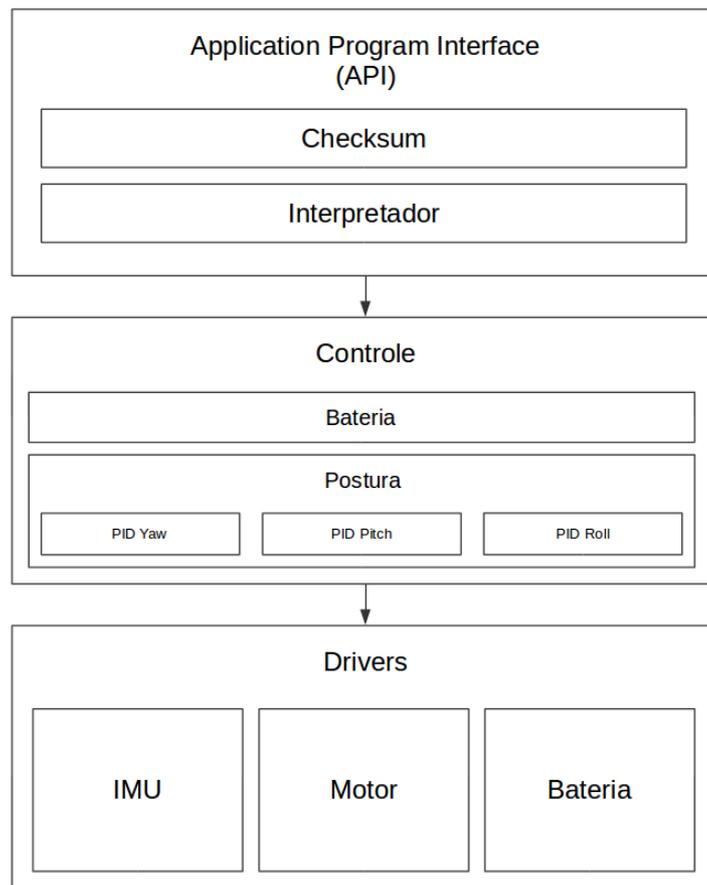


Figura 4 – Design da arquitetura em camadas proposta

6.1 Camada de drivers

A camada de drivers faz a comunicação direta com os periféricos conectados ao micro controlador, ela é responsável por adquirir dados sensoriais e comandar os atuadores do sistema.

6.1.1 Classe IMU

A classe IMU é responsável por adquirir os dados sensoriais do periférico MPU6050, onde se fazem presentes o acelerômetro e giroscópio. Os dados adquiridos destes sensores serão convertidos para um formato padrão adotado pelo projeto para então serem enviados a camada requisitante.

6.1.2 Classe Motor

A classe motor é responsável por comandar diretamente os atuadores do tipo motor de corrente contínua, de forma individual. Neste projeto ela faz uso de portas *Pulse Width Modulation* (PWM) existentes na plataforma Arduino para enviar sinais analógicos que determinam a velocidade dos motores.

6.1.3 Classe Bateria

A classe Bateria é responsável por realizar leituras referentes a tensão fornecida pela fonte de energia utilizada pelo protótipo, e reenviar estes dados a camada requisitante.

6.2 Camada de controle

A camada de controle é responsável por processar os cálculos e algoritmos que coordenam os sensores e atuadores do protótipo. Essa camada se utiliza da camada de drivers para realizar as leituras dos sensores e controlar os atuadores.

6.2.1 Classe Bateria

No protótipo utilizado para este trabalho foi implementada a classe de controle da bateria. Esta classe é responsável por realizar a leitura de tensão fornecida pela fonte de energia por meio da camada de drivers. Os valores de tensão fornecidos pela camada de drivers são utilizados para o controle dos atuadores, assim o sistema considera a quantidade de carga da bateria durante o comando dos mesmos.

6.2.2 Classe Postura

A classe Postura é responsável por processar os algoritmos de controle da postura inercial. Essa classe faz uso da camada de drivers, requisitando dados sobre a postura corrente do quadrotor, que são utilizados para alimentar algoritmos de controle PID. Nos controladores PID o valor de erro é calculado a partir do sinal emitido pela camada de drivers que é subtraído do sinal de *feedback*,

também emitido pela camada de drivers, para gerar o sinal de erro, onde é então enviado para os componentes do controle, os sinais resultantes são somados e o termo final é utilizado para o controle do sistema.

Os algoritmos de controle PID usados pela classe Postura são: i) PID *Yaw*, que corrige o ângulo do quadrotor em torno do eixo z, ii) PID *Pitch*, que corrige o ângulo do quadrotor em torno do eixo y, iii) PID *Roll*, que corrige o ângulo do quadrotor em torno do eixo x.

6.3 Camada de API

A camada de API é responsável por receber e interpretar comandos externos, e então redirecioná-los a camada de controle, ela também é responsável por enviar dados requisitados do sistema.

6.3.1 Classe Checksum

A classe *checksum* é responsável por verificar a integridade dos dados que chegam à camada de API. Nela é implementada um algoritmo de *checksum*, onde é calculada a soma de verificação ao receber os dados. Se o valor obtido é o mesmo do *byte* de verificação, as informações não sofreram alterações e, portanto, não estão corrompidas.

6.3.2 Classe Interpretador

A classe Interpretador é responsável por interpretar os comandos externos. Se o comando recebido for interpretado com sucesso a classe interpretador então aciona as classes e funções necessárias da camada de controle para que o comando seja executado.

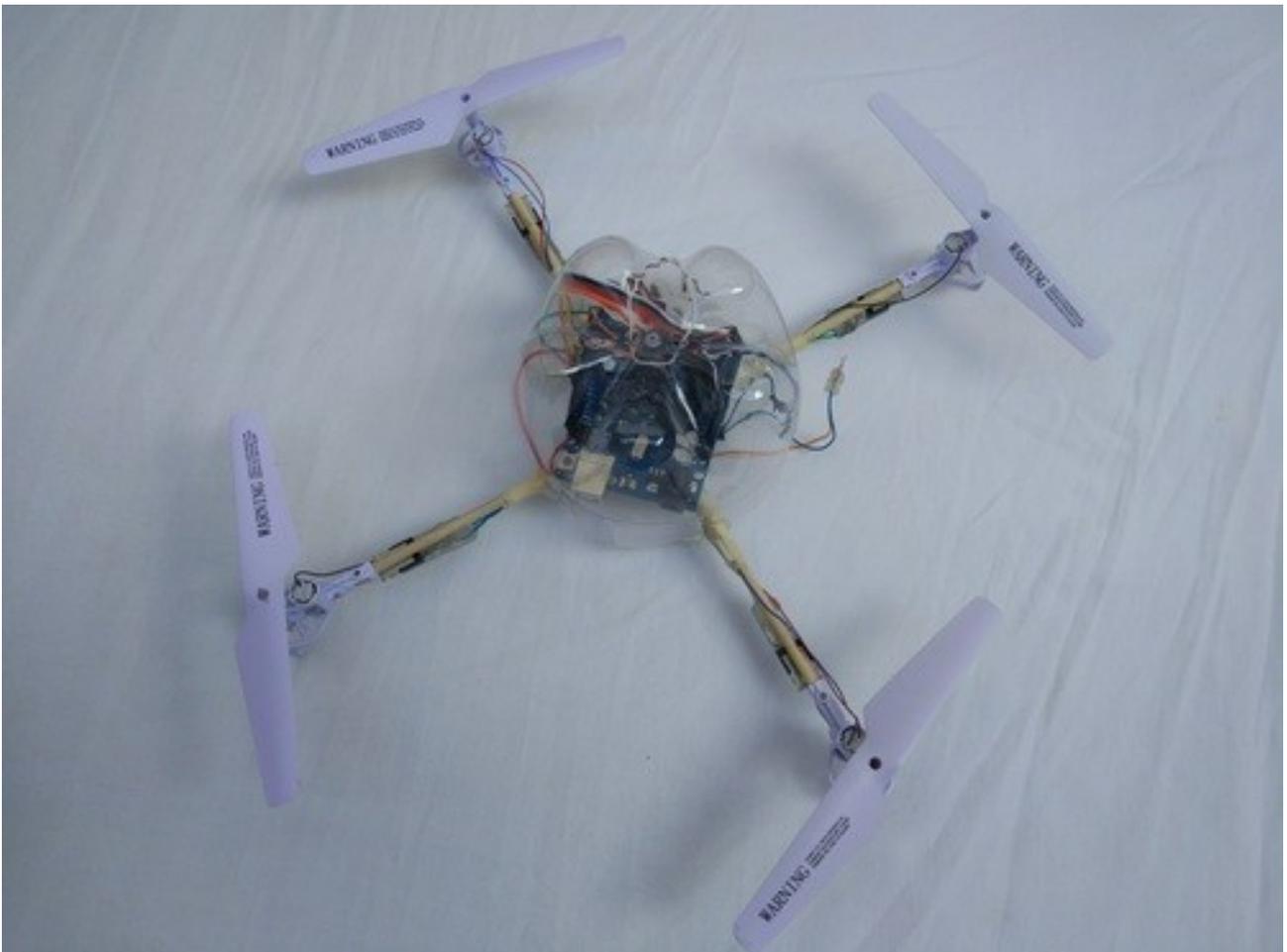
7 Resultados

Baseando-se nos conceitos apresentados neste trabalho foi elaborado um diagrama de classes que segue os padrões da arquitetura proposta, mostrado na Figura 8.

O programa gerado a partir do diagrama de classe ilustrado na Figura 8 foi desenvolvido usando a linguagem C++ e compilado através do compilador GCC versão 5.4.0.

A validação dos recursos previstos na arquitetura proposta foi realizada com a adoção da seguinte estrutura: i) configurado o ECAD para receber e enviar os sinais equivalentes aos comandos que deverão ser recebidos pelo sistema programado para UCP, ii) elaborado o programa para aplicar conceitos mostrados na Figura 4 e atender os passos do fluxograma analítico mostrados na Figura 3 seguindo o diagrama de classes mostrado na Figura 8, iii) A execução dos testes foi realizada avaliando o comportamento do protótipo durante voo.

Durante o funcionamento do sistema o protótipo realizou o voo executando todos os controles de estabilidades programados e foi capaz de checar e interpretar os comandos provenientes do ECAD.



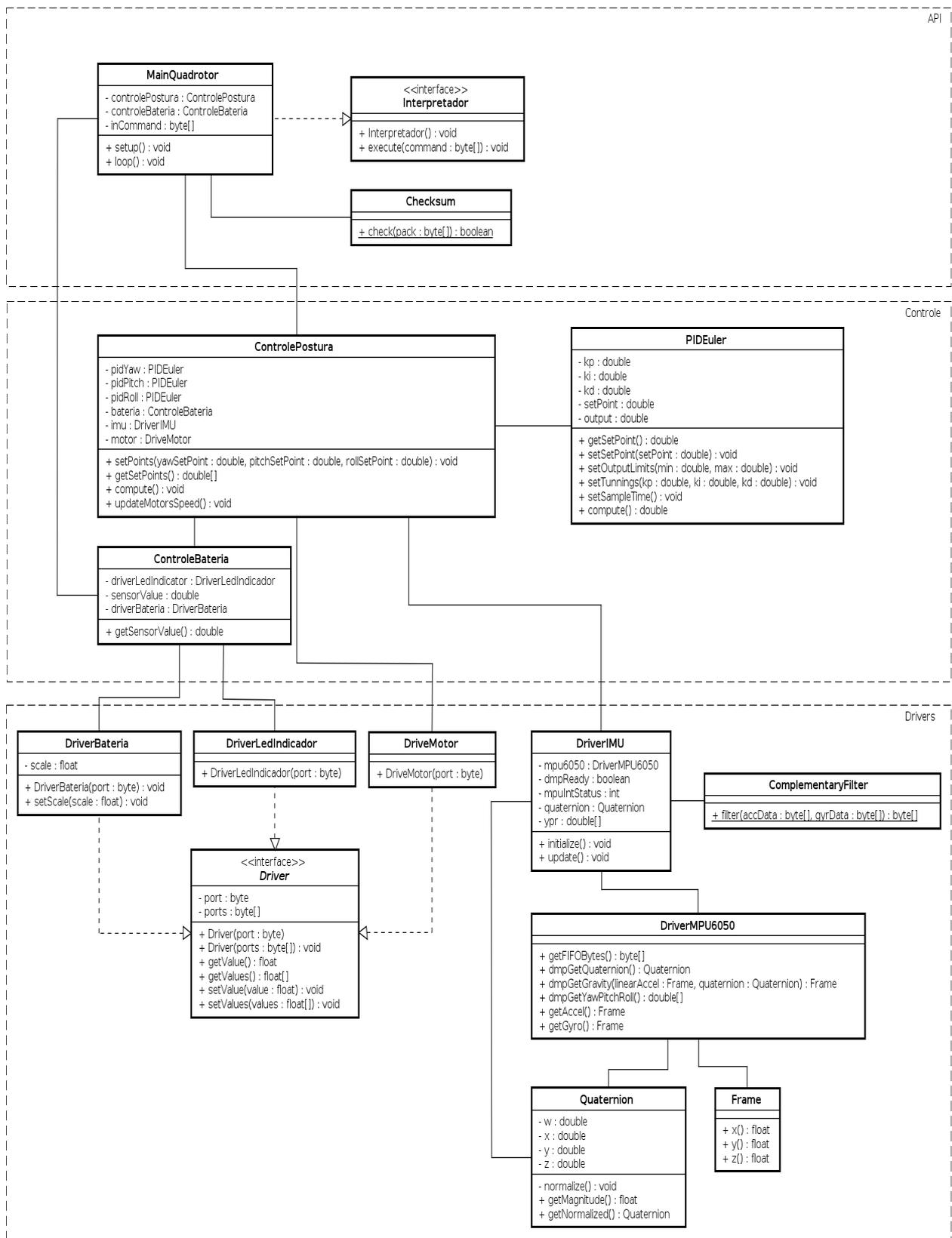


Figura 8 – Diagrama de classes do programa principal

8 Discussão

O presente trabalho apresenta uma proposta de design de uma arquitetura lógica capaz de realizar os controles operacionais de *drones* quadrotoros.

Esta arquitetura é uma instância da arquitetura em camadas, que difere do modelo de Sanchez (2016) utilizando menos camadas e, assim, condensando e generalizando as classes e funcionalidades similares, o que torna este sistema de fácil implementação e melhor difusão por comunidades open source.

Esse sistema também se difere do modelo proposto por Omari (2013), pois separa os algoritmos de controle dos algoritmos de interpretação de comando externo em camadas diferentes o que facilita sua implementação e principalmente sua organização e manutenibilidade tendo em vista que suas funções são de fato completamente diferentes o que justifica sua separação em camadas especialistas.

9 Conclusão

O objetivo de propor e avaliar uma arquitetura de programação elaborada para operar um drone quadrotor foi atingido neste trabalho.

Os resultados positivos atingidos nos testes práticos realizados com o protótipo sugerem que a arquitetura proposta permite a implementação de camadas especialistas, escalabilidade de recursos e fácil manutenção do software.

As estruturas utilizadas pela arquitetura assim como as classes elaboradas durante o teste prático poderão ser adotadas em outros tipos de aplicação com poucas ou nenhuma modificação.

Referências bibliográficas

ALEXIS, Kostas; NIKOLAKOPOULOS, George; TZES, Anthony. Model predictive quadrotor control: attitude, altitude and position experimental studies. *IET Control Theory & Applications*, v. 6, n. 12, p. 1812-1827, 2012.

BENIĆ, Zoran; PILJEK, Petar; KOTARSKI, Denis. Mathematical modelling of unmanned aerial vehicles with four rotors. *Interdisciplinary Description of Complex Systems*, v. 14, n. 1, p. 88-100, 2016.

BOLDEA, Ion; NASAR, Syed A. Linear electric actuators and generators. In: *Electric Machines and Drives Conference Record, 1997. IEEE International. IEEE, 1997. p. MA1/1.1-MA1/1.5.*

CAVALCANTE, Marisa Almeida; TAVOLARO, Cristiane Rodrigues Caetano; MOLISANI, Elio. Physics with Arduino for beginners. *Revista Brasileira de Ensino de Física*, v. 33, n. 4, p. 4503-4503, 2011.

CHAMBERLIN, Tim. *Quadcopter Dynamics: Modeling and Stabilizing Quadrotor Aircraft*. 2016. Tese de Doutorado. Bates College.

FUX, Samuel. *Development of a planar low cost Inertial Measurement Unit for UAVs and MAVs*. Swiss federal Institute of Technology Zurich, 2008.

GARLAN, David; SHAW, Mary. An introduction to software architecture. *ADVANCES IN SOFTWARE ENGINEERING AND KNOWLEDGE ENGINEERING*, v. 1, n. 3.4, 1993.

GOODMAN, Jacob M. et al. System and mathematical modeling of quadrotor dynamics. In: *SPIE Defense+ Security*. International Society for Optics and Photonics, 2015. p. 94680R-94680R-11.

HIGGINS, Walter T. A comparison of complementary and Kalman filtering. *IEEE Transactions on Aerospace and Electronic Systems*, v. 11, n. 3, p. 321-325, 1975.

KAYA, Derya; KUTAY, Ali Türker. Modeling and Simulation of a Quadrotor using Curve Fitting Method. In: AIAA Atmospheric Flight Mechanics Conference. 2015. p. 2706.

LENGERKE, Omar; GONZALEZ, Hernan; BELTRÁN, Sergio Andrés. Control PID de altura de un Quadrotor. Revista Colombiana de Computación-RCC, v. 12, n. 2, 2013.

MANKAR, Jayant et al. Review of I2C protocol. International Journal, v. 2, n. 1, 2014.

PAIVA, Ernesto A. et al. Modeling and PID cascade control of a Quadcopter for trajectory tracking. In: 2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON). IEEE, 2015. p. 809-815.

POUNDS, Paul; MAHONY, Robert; CORKE, Peter. Modelling and control of a large quadrotor robot. Control Engineering Practice, v. 18, n. 7, p. 691-699, 2010.

SÁ, Rejane Cavalcante et al. Construction and PID control for stability of an unmanned aerial vehicle of the type quadrotor. In: Robotics Symposium and Competition (LARS/LARC), 2013 Latin American. IEEE, 2013. p. 95-99.

SHAW, Mary; GARLAN, David. Software architecture: perspectives on an emerging discipline. Englewood Cliffs: Prentice Hall, 1996.

TAHIR, Zaid et al. State Space System Modeling of a Quad Copter UAV. Indian Journal of Science and Technology, v. 9, n. 27, 2016.

WESCOTT, Tim. PID without a PhD. Embedded Systems Programming, v. 13, n. 11, p. 1-7, 2000.

_____, MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking™ Devices. Disponível em: <<https://www.invensense.com/products/motion-tracking/6-axis/mpu-6050/>>. Acesso em: 1 de novembro de 2016.

Anexo

Diretrizes Para Autores

Os trabalhos devem ser redigidos em português, com uso obrigatório da norma culta. Os nomes dos autores, bem como a afiliação institucional de cada um, devem ser inseridos nos campos adequados a serem preenchidos durante a submissão e devem aparecer no arquivo. A Revista Eletrônica de Ciências Humanas sugere que o número máximo de autores por artigo seja 6 (seis). Artigos com número superior a 6 (seis) serão considerados exceções e avaliados pelo Conselho Editorial que poderá solicitar a adequação. **Pesquisas feitas com seres humanos e animais devem, obrigatoriamente, citar a aprovação da pesquisa pelo respectivo Comitê de Ética, citando o protocolo de aprovação.** O não atendimento de tal proposta pode implicar em recusa de sua publicação. Da mesma forma, o plágio implicará na recusa do trabalho.

Os autores dos artigos aceitos poderão solicitar a tradução do artigo para língua inglesa aos tradutores indicados pela revista e reenviar. Os custos com a tradução serão de responsabilidade dos autores.

O periódico disponibilizará aos leitores o conteúdo digital em ambos os idiomas, português e inglês.

O uso da norma culta da Língua Portuguesa e a obediência às normas da Revista são de total responsabilidade dos autores. A não obediência a esses critérios implicará na recusa imediata do trabalho.

Apresentação Do Material

Sugere-se um número máximo de 20 páginas, incluindo referências, figuras, tabelas e quadros. Os textos devem ser digitados em **Fonte Times New Roman, tamanho 12**,

espaçamento 1,5, justificado, exceto Resumo e Abstract. Devem ser colocadas margens de 2 cm em cada lado.

As Figuras: gráficos, imagens, desenhos e esquemas deverão estar inseridas no texto, apresentar boa qualidade, estar em formato JPEG, com resolução de 300dpi com 15cm x 10cm. O número de figuras deve ser apenas o necessário à compreensão do trabalho. Não serão aceitas imagens digitais artificialmente 'aumentadas' em programas computacionais de edição de imagens. As figuras devem ser numeradas em algarismos arábicos segundo a ordem em que aparecem e suas legendas devem estar logo abaixo.

Tabelas e Quadros: deverão ser numerados consecutivamente com algarismos arábicos e encabeçados pelo título. As tabelas e os quadros devem estar inseridos no texto. Não serão admitidas as tabelas e quadros inseridos como Figuras.

Títulos de tabelas e quadro e legendas de figuras deverão ser escritos em tamanho 11 e com espaço simples entre linhas.

Citação no texto: deve-se seguir as Normas da ABNT (NBR 10520, 2003). As citações deverão aparecer no texto, seguidas pelo ano de publicação. As chamadas pelo sobrenome do autor, pela instituição responsável ou título podem ser: a) incluídas na sentença: sobrenome (ano). Ex.: Gomes, Faria e Esper (2006) ou b) entre parênteses: (SOBRENOME, ano). Ex.: (GOMES; FARIA; ESPER, 2006). Quando se tratar de citação direta (transcrição literal), indicar, após o ano, a página de onde o texto foi extraído. O trecho transcrito deverá estar entre aspas quando ocupar até três linhas. As citações diretas com mais de três linhas devem ser destacadas com recuo de 4 cm da margem esquerda, ser escritas com letra menor que a do texto utilizado, com espaçamento entre linhas menor do que o utilizado no texto e sem as aspas. Citações indiretas de vários documentos simultaneamente devem constar em ordem alfabética (como nas referências). Citação de citação: autor citado (ano apud AUTOR, ano). Deve-se fazer a referência do autor lido. Ex.: Pádua (1996 apud FERNANDES, 2012, p. 5) salienta que “[...] pesquisa é toda atividade voltada para a solução de problemas [...]”.

Teses, dissertações e monografias, solicitamos que sejam utilizados apenas documentos dos **últimos três anos** e quando não houver o respectivo artigo científico publicado em periódico. Esse tipo de referência deve, obrigatoriamente, **apresentar o link** que remeta ao cadastro nacional de teses da CAPES e aos bancos locais das universidades que publicam esses documentos no formato pdf.

Grafia de termos científicos, comerciais, unidades de medida e palavras estrangeiras: os termos científicos devem ser grafados por extenso, em vez de seus correspondentes simbólicos abreviados. Para unidades de medida, deve-se utilizar o Sistema Internacional de Unidades. Palavras em outras línguas devem ser evitadas nos textos em português, utilizar preferentemente a sua tradução. Na impossibilidade, os termos estrangeiros devem ser grafados em itálico. Toda abreviatura ou sigla deve ser escrita por extenso na primeira vez em que aparecer no texto.

Estrutura Do Artigo

PESQUISAS ORIGINAIS devem ter no máximo 20 páginas com até 40 citações; organizar da seguinte forma:

Título em português: caixa alta, centrado, negrito, conciso, com um máximo de 25 palavras;

Título em inglês (obrigatório): caixa alta, centrado. Versão do título em português;

Autor(es): O(s) nome(s) completo(s) do(s) autor(es) e seus títulos e afiliações à Sociedade ou Instituições. Indicar com asterisco o autor de correspondência. Ao final das afiliações fornecer o e-mail do autor de correspondência.

Resumo: parágrafo único sem deslocamento, fonte tamanho 11, espaço 1, justificado, contendo entre 150 e 250 palavras. Deve conter a apresentação concisa de cada parte do trabalho, abordando objetivo(s), método, resultados e conclusões. Deve ser escrito sequencialmente, sem subdivisões. Não deve conter símbolos e contrações que não sejam de uso corrente nem fórmulas, equações, diagramas;

Palavras-chave: de 3 a 5 palavras-chave, iniciadas por letra maiúscula, separadas e finalizadas por ponto.

Abstract (obrigatório): fonte tamanho 11, espaço 1, justificado, deve ser a tradução literal do resumo;

Keywords: a apresentação deverá ser a mesma das Palavras-chave em Português.

Introdução: deve apresentar o assunto a ser tratado, fornecer ao leitor os antecedentes que justificam o trabalho, incluir informações sobre a natureza e importância do problema, sua relação com outros estudos sobre o mesmo assunto, suas limitações. Essa seção deve representar a essência do pensamento do pesquisador em relação ao assunto estudado e apresentar o que existe de mais

significante na literatura científica. Os objetivos da pesquisa devem figurar como o último parágrafo desse item.

Método: destina-se a expor os meios dos quais o autor se valeu para a execução do trabalho. Pode ser redigido em corpo único ou dividido em subseções. Especificar tipo e origem de produtos e equipamentos utilizados. Citar as fontes que serviram como referência para o método escolhido.

Pesquisas feitas com seres humanos e animais devem, obrigatoriamente, citar a aprovação da pesquisa pelo respectivo Comitê de Ética, citando o protocolo de aprovação.

Resultados: Nesta seção o autor irá expor o obtido em suas observações. Os resultados poderão estar expressos em quadros, tabelas, figuras (gráficos e imagens). Os dados expressos não devem ser repetidos em mais de um tipo de ilustração.

Discussão: O autor, ao tempo que justifica os meios que usou para a obtenção dos resultados, deve contrastar esses com os constantes da literatura pertinente; estabelecer relações entre causas e efeitos; apontar as generalizações e os princípios básicos, que tenham comprovações nas observações experimentais; esclarecer as exceções, modificações e contradições das hipóteses, teorias e princípios diretamente relacionados com o trabalho realizado; indicar as aplicações teóricas ou práticas dos resultados obtidos, bem como, suas limitações; elaborar, quando possível, uma teoria para explicar certas observações ou resultados obtidos; sugerir, quando for o caso, novas pesquisas, tendo em vista a experiência adquirida no desenvolvimento do trabalho e visando a sua complementação.

Conclusões: Devem ter por base o texto e expressar com lógica e simplicidade o que foi demonstrado com a pesquisa, não se permitindo deduções. Devem responder à proposição.

Agradecimentos (opcionais): O autor deve agradecer às fontes de fomentos e àqueles que contribuíram efetivamente para a realização do trabalho. Agradecimento a suporte técnico deve ser feito em parágrafo separado.

Referências (e não bibliografia): Espaço simples entre linhas e duplo entre uma referência e a próxima. As referências devem ser numeradas na ordem em que aparecem no texto. A lista completa de referências, no final do artigo, deve estar de acordo com as normas da ABNT (NBR 6023, 2003). Quando a obra tiver até três autores, todos devem ser citados. Mais de três autores, indicar o primeiro, seguido de et al. Alguns exemplos:

Artigo publicado em periódico:

LUDKE, M.; CRUZ, G. B. dos. Aproximando universidade e escola de educação básica pela pesquisa. **Caderno de pesquisa**, São Paulo, v. 35, n. 125, p. 81-109, maio/ago. 2005.

Artigo publicado em periódico em formato eletrônico:

SILVA JUNIOR, N. A. da. Satisfação no trabalho: um estudo entre os funcionários dos hotéis de João Pessoa. **Psico-USF**, Itatiba, v. 6, n. 1, p. 47-57, jun. 2001. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1413-82712001000100007&lng=pt&nrm=iso>. Acesso em: 13 jul 2015.

Livro (como um todo)

MENDONÇA, L. G. et al. **Matemática financeira**. 10. ed. Rio de Janeiro: Editora FGV, 2010.

Capítulo de livro

MARTÍN. E.; SOLÉ, I. A aprendizagem significativa e a teoria da assimilação. In: COLL, C.; MARCHESI, A.; PALACIOS, J. (Org.). **Desenvolvimento psicológico e educação: psicologia da educação escolar**. 2. ed. Porto Alegre: Artmed, 2008. cap. 3, p. 60-80.

Artigos De Revisão

Poderão ser aceitos para submissão, desde que abordem temas de interesse, atualizados. Devem ser elaborados por pesquisadores com experiência no campo em questão ou por especialistas de reconhecido saber. Devem ter até 20 páginas, incluindo resumos, tabelas, quadros, figuras e referências. As tabelas, quadros e figuras limitadas a 06 no conjunto, devem incluir apenas os dados imprescindíveis. As figuras não devem repetir dados já descritos em tabelas. As referências bibliográficas devem ser limitadas a 60. Deve-se evitar a inclusão de número excessivo de referências numa mesma citação.

Devem conter: título em português e inglês, autores e afiliações, resumo e abstract (de 150 a 250 palavras), palavras-chave/keywords, introdução, método (como nos artigos de pesquisas originais) considerações finais (neste item serão retomadas as diferentes colocações dos autores estudados de maneira a conduzir a um fechamento, porém, não havendo conclusões definitivas), agradecimentos (caso necessário), referências.

Ou, em caso de artigos de revisão de literatura contendo metanálise, depois do item método deverá ser apresentado o item resultados (contendo a metanálise) e as conclusões.

Autorizo cópia total ou parcial desta obra,
apenas para fins de estudo e pesquisa,
sendo expressamente vedado qualquer
tipo de reprodução para fins comerciais
sem prévia autorização específica do
autor. Autorizo também a divulgação do
arquivo no formato PDF no banco de
monografias da Biblioteca institucional.
Rodrigo Reis Alves
Pindamhangaba, 12/2016.