

# Uso Consciente da Dimerização e da Temperatura de Cor na Automação de Lâmpadas Inteligentes

Gabriel B. Takahashi \* Fernanda Q. Rossi \*\* André M. Oliveira \*\*\*

\* Instituto de Ciência e Tecnologia, Universidade Federal de São Paulo, SP, (e-mail: gabriel.borin@unifesp.br).

\*\* Instituto de Ciência e Tecnologia, Universidade Federal de São Paulo, SP, (e-mail: rossi.fernanda@unifesp.br)

\*\*\* Instituto de Ciência e Tecnologia, Universidade Federal de São Paulo, SP, (e-mail: andre.marcorin@unifesp.br)

---

**Abstract:** Smart lamps actually have dimmable LEDs, which are feasible to vary their luminous intensity and, furthermore, as they have two high power LEDs (one is warm white and the other one is cold white), it becomes possible to control their Correlated Color Temperature. This review proposes an automatic activation of smart lamps as it gets on when ambience becomes dark, then automatic dimming until the lights are completely turned off at the end of the day. At the end, the conscious use of the Correlated Color Temperature through an automation suggesting the reduction of visible light frequency with high-energy in periods leading up to the sleep phase, in order to improve human welfare. The behavior of the automatic lighting system is presented in experimental tests for different situations in the ambience.

**Resumo:** As lâmpadas inteligentes possuem LEDs dimerizáveis, em que é possível regular sua intensidade luminosa e, além disso, como possuem dois LEDs de alta potência (um branco quente e um branco frio), é possível controlar sua Temperatura de Cor Correlacionada. Este trabalho propõe um acionamento automático das lâmpadas inteligentes conforme o escurecer, a dimerização automática até o desligamento total das luzes ao chegar no final do dia. Por fim, o uso consciente da Temperatura de Cor Correlacionada através de uma automação sugerindo a diminuição das frequências de luz de alta energia em períodos que antecedem o sono, a fim de melhorar o bem estar humano. O comportamento do sistema automático de iluminação é apresentado em testes experimentais para diferentes situações no ambiente.

**Keywords:** Smart lamps; luminous intensity; Correlated Color Temperature; dimming; automation.

**Palavras-chaves:** Lâmpadas inteligentes; intensidade luminosa; Temperatura de Cor Correlacionada; dimerização; automação.

---

## 1. INTRODUÇÃO

Com a evolução da tecnologia e o avanço da automação residencial, o uso de lâmpadas inteligentes (*smart bulbs*) nas residências vêm crescendo, segundo Statista (2022) espera-se um aumento de pouco mais de 2% em 2020 para quase 8% em 2025. Este tipo de lâmpada possui dois LEDs (*Light Emitting Diodes*) de alta potência, um branco frio (branco azulado) e um branco quente (branco amareado), e um LED RGB (*Red, Green, Blue*) de baixa potência, diferente das lâmpadas LEDs convencionais, que possuem apenas um LED de alta potência. Desta forma, a lâmpada inteligente permite regular sua intensidade luminosa (dimerização) e sua temperatura de cor, vantagens em relação à lâmpada convencional que tem apenas dois estados, ligado e desligado.

Estudos mostram a importância da temperatura de cor da iluminação. De acordo com Kulve et al. (2018), existe uma temperatura de cor recomendada para cada ambiente, de forma a promover através das cores um aumento do estado de alerta ou do estado relaxado. Em situações em que um

melhor desempenho cognitivo é desejado, recomenda-se a luz azulada (luz branco frio). Por outro lado, segundo Kim and Lee (2019), recomenda-se uma menor incidência de raios azuis em períodos que antecedem o sono, para evitar distúrbios de sono. Já em Viola et al. (2008) e Chellappa et al. (2011), é relatado que diferentes temperaturas de cor impactam significativamente não só no desempenho cognitivo, mas também na fisiologia circadiana, a qual influencia em funções corporais, como digestão, sono, temperatura corporal, apetite e hormônios.

Vários trabalhos vêm buscando aprimorar a iluminação residencial e do ambiente de trabalho. Em Buso and Spiazzi (2011), foi apresentado um controlador para regular o fluxo luminoso e a temperatura de cor de uma lâmpada branca de estado sólido, que possui três matrizes de LED (vermelho, verde e branco frio), por meio do ajuste de cada matriz de LED por dimerização PWM (*Pulse Width Modulation*). Em Byun et al. (2013), foi proposto um sistema inteligente, com sensores e LED, que ajusta a intensidade mínima de luz necessária para cada ambiente de forma autônoma, visando reduzir o consumo de energia.

Em Muhamad et al. (2018), foi desenvolvido um sistema de controle inteligente da dimerização de luz e temperatura de cor correlacionada utilizando um aplicativo *Android*, de forma a fornecer conforto ao usuário e economizar energia. Adicionalmente, em Lee et al. (2016), um método de controle não linear foi proposto para controlar a luminosidade e a temperatura de cor de uma lâmpada LED com ajuste bicolor, de forma mais precisa e robusta às variações da temperatura do ambiente, à interferência da luz ambiente e ao envelhecimento do LED.

Atualmente, existem aplicativos móveis disponíveis para controlar a iluminação das *smart* lâmpadas, como por exemplo lâmpadas das marcas Elgin, Positivo e Xiaomi, onde permitem a execução de determinadas tarefas de forma automática. Por exemplo, tarefa de ligar ou desligar o dispositivo, definição do modo de cor (LED branco ou colorido), da intensidade de brilho, da temperatura de cor (branco quente ou branco frio), e a opção de desligamento automático. Estas tarefas podem ser em função de certas condições, como alteração no clima local, ou por meio do agendamento em um horário predefinido. No entanto, para o usuário realizar a dimerização da luz ao longo do dia, torna-se necessário criar  $n$  tarefas no aplicativo, sendo  $n$  o número de transições para a diminuição da intensidade luminosa gradualmente. Além disso, na execução de uma tarefa, a transição da luz ocorre de forma abrupta. Dessa forma, torna-se uma tarefa difícil e limitada para o usuário usufruir dos benefícios de uma lâmpada inteligente.

Assim, neste trabalho é proposto o controle automático da intensidade luminosa e da temperatura de cor deste dispositivo, com transições de forma suave. Com este propósito, foi criado um sistema de iluminação empregando um LED RGB e um microcontrolador, para representar o funcionamento de uma lâmpada inteligente. De forma a diminuir o consumo de luz e ter maior satisfação do usuário, foi adicionado ao sistema um sensor luminoso. Além disso, foi desenvolvido uma aplicação *web* para realizar o controle automático da lâmpada inteligente. Dessa forma, é possível realizar uma automação de forma mais intuitiva e simples, proporcionando maior conforto ao usuário.

O restante do artigo é organizado conforme descrito a seguir. Na Seção 2, são definidos os conceitos básicos e componentes empregados no sistema automático de iluminação. O desenvolvimento do projeto é descrito na Seção 3. Implementações experimentais e resultados são apresentados na Seção 4. Por fim, na Seção 5, apresenta-se a conclusão do trabalho.

## 2. SISTEMA DE ILUMINAÇÃO

As grandezas de iluminação, os componentes eletrônicos e conceitos adicionais relacionados a sistemas embarcados, utilizados para elaboração do sistema automático de iluminação, serão apresentados nesta seção, para uma melhor compreensão do trabalho desenvolvido. Inicialmente, na Subseção 2.1, serão descritas as grandezas de iluminação tratadas. Já os componentes utilizados e os conceitos adicionais serão descritos na Subseção 2.2.

### 2.1 Grandezas de Iluminação

Neste trabalho, foram tratadas as seguintes grandezas de iluminação: a intensidade e a temperatura de cor correlacionada.

A intensidade está relacionada ao fluxo de luz. É o fluxo de uma fonte pontual contido em um pequeno volume angular, que pode variar conforme sua direção (Arecchi et al. (2007)).

A temperatura de cor é uma representação unidimensional de diferentes tonalidades de fontes de luz branca, geralmente expressa em Kelvin (K) (Ákos et al. (2001)). É definida com base na cromaticidade, a qual quantifica a cor percebida de uma fonte luminosa, que pode ser caracterizada por duas coordenadas cromáticas. Este conceito é utilizado para lâmpadas incandescentes (Arecchi et al. (2007)). Já para luzes brancas que não possuem essas coordenadas de cromaticidade, é utilizado o conceito de temperatura de cor correlacionada (*Correlated Color Temperature* ou CCT), também expressa em Kelvin (K). Com base na métrica CCT pode ser feita a percepção da cor emitida por luzes fluorescentes e LEDs (Arecchi et al. (2007)).

### 2.2 Sistema Automático de Iluminação

Para a elaboração do sistema automático de iluminação, foram utilizados os seguintes componentes: LED RGB, fotoresistor (sensor) e microcontrolador.

O LED RGB possui três LEDs em um mesmo encapsulamento: um vermelho (*Red*), um verde (*Green*) e um azul (*Blue*). Com isso, permite a emissão de diversas cores de luz.

O fotoresistor, ou resistor dependente de luz (LDR, *Light Dependent Resistor*), é um sensor de luminosidade, que varia sua resistência de forma inversamente proporcional à intensidade de luz incidindo sobre ele.

Já o microcontrolador é um circuito integrado que contém processador, memória e periféricos, como conversor A/D (Analogico/Digital), gerador de PWM (*Pulse Width Modulation*), temporizadores e interfaces de comunicação externa. Este dispositivo é utilizado para controlar uma tarefa ou processo específico. Geralmente, é responsável por receber e interpretar dados de um sensor, e retornar comandos para controle do sistema. Por ser um sistema computacional compacto e de custo acessível, tem sido muito empregado na área de controle, automação e sistemas embarcados.

A seguir são descritos os periféricos de um microcontrolador utilizados neste trabalho.

Um conversor analógico digital (A/D) é um circuito eletrônico que transforma um sinal analógico em um sinal digital. Como geralmente os microcontroladores recebem sinais analógicos de sensores, utilizam este tipo de periférico como interface de entrada destes sinais, de forma a ter o sinal codificado em formato digital (Almeida et al. (2017)).

O gerador de PWM permite o fornecimento de uma saída PWM ao circuito, ou seja, de um sinal periódico com forma

similar à onda quadrada, mas que pode ter a variação da duração do pulso e, com isto, a variação de seu valor médio. Isso permite, por exemplo, controlar a luminosidade de lâmpadas e a velocidade de motores CC. A porcentagem do tempo do pulso do sinal PWM em relação à duração do ciclo completo (período), é chamado de *duty cycle*. A potência do sinal PWM varia na mesma proporção da largura do pulso. Quanto maior a largura do pulso (maior *duty cycle*), maior a potência (Almeida et al. (2017)).

Uma das interfaces de comunicação externa é a comunicação serial sem fio (*Wireless*), que vem sendo muito utilizada. Este tipo de comunicação permite a transmissão de dados pelo ar (Almeida et al. (2017)). Para esta forma de comunicação, existem protocolos como o *Bluetooth*, o *ZigBee* e a rede *Wi-Fi*. Neste trabalho, será utilizada a comunicação *Wi-Fi*, protocolo no padrão IEEE 802.11. Este tipo de comunicação transmite os dados através da propagação de sinais de radiofrequência pelo ar (Kurose et al. (2007)).

Para realizar a comunicação entre duas ou mais entidades remotas pela Internet são empregados protocolos de rede, como, por exemplo, HTTP (*Hypertext Transfer Protocol*) e TCP (*Transmission Control Protocol*). O protocolo HTTP é utilizado na camada de aplicação da Web, ocorrendo em um programa cliente e um programa servidor. O cliente é responsável por iniciar a conexão, com a requisição de uma página *web* ao servidor. O servidor, então, retorna uma resposta com o objeto desejado ao cliente, finalizando a conexão, sem armazenar informações sobre o cliente. Este protocolo define como as requisições do cliente são feitas e como os servidores fazem a transferência de dados. Para isso, o TCP é utilizado como protocolo da camada de transporte subjacente. Dessa forma, o cliente HTTP faz a inicialização de uma conexão TCP com o servidor. Estabelecida esta conexão, os processos do navegador e do servidor acessam o TCP através da interface de *socket*. Essa interface *socket*, no lado cliente, é a porta entre o processo cliente e a conexão TCP. Já no lado servidor, esta interface é a porta entre o processo servidor e a conexão TCP (Kurose et al. (2007)).

### 3. CONTROLE DE INTENSIDADE LUMINOSA E TEMPERATURA DE COR

Para o desenvolvimento do sistema, foi projetada uma arquitetura (representada na Figura 1) capaz de simular uma lâmpada inteligente, assim como a captura da intensidade luminosa ambiente. Na simulação da lâmpada inteligente, utilizou-se um circuito contendo um LED RGB conectado a um ESP8266, dessa forma, é possível realizar o controle do LED RGB à distância através de uma aplicação *web* ou de um aplicativo móvel. Do outro lado, foi projetado um circuito elétrico passivo contendo um LDR responsável por capturar a intensidade luminosa ambiente e se comunicar com a lâmpada inteligente simulada, utilizando um segundo ESP8266.

Além disso, foi implementada uma aplicação *web* responsável principalmente pela automação da Temperatura de Cor definida pelo usuário. Esta aplicação é hospedada no ESP8266 onde está conectado o LED RGB e, para uma melhor compreensão, este microcontrolador será chamado de servidor, pois recebe tanto os dados da Temperatura

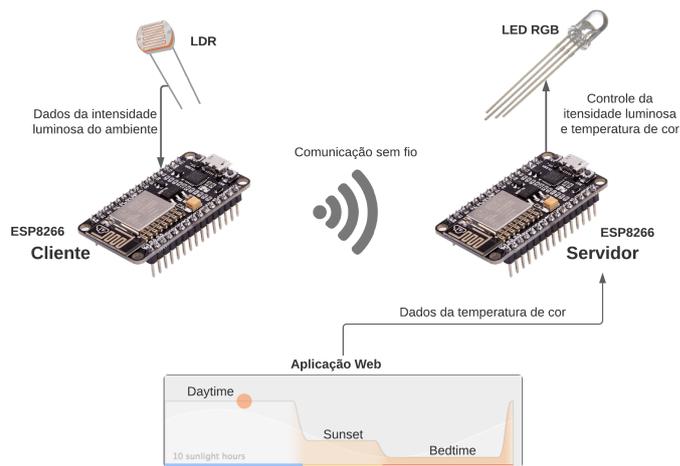


Figura 1. Esquemático da arquitetura do sistema.

de Cor oriundos da aplicação *web*, como também recebe informações da intensidade luminosa ambiente vindos do outro ESP8266, denominado cliente.

Nas Subseções 3.1 e 3.2, são explicados como funcionam os sistemas automáticos da intensidade luminosa e da Temperatura de Cor.

#### 3.1 Sistema Automático de Intensidade Luminosa

O sistema automático da intensidade luminosa baseia-se em capturar a intensidade luminosa ambiente utilizando um fotorresistor, calcular a média móvel dos últimos valores armazenados e transmitir, por comunicação *wireless*, a média móvel calculada para o servidor, responsável pelo controle do LED RGB.

Para isso, foi projetado um circuito elétrico passivo composto por um resistor de 10 k $\Omega$ , um LDR e uma fonte de tensão de 3.3 V, representados na Figura 2. A saída ADO do circuito é conectada à entrada analógica do ESP8266, que possui um conversor analógico digital de 10 *bits* de resolução, resultando em um valor digital de 0 a 1023.

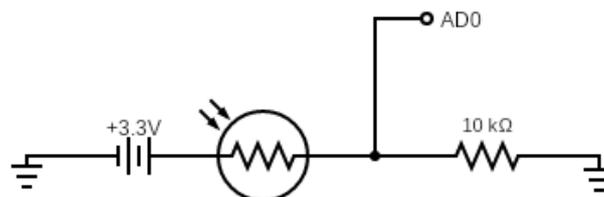


Figura 2. Circuito passivo para o LDR.

Com o intuito de manter o sistema robusto a perturbações luminosas impermanentes como sombras ou mudanças bruscas de luminosidade ambiente, foi utilizado o filtro de média móvel, assim o sistema responderá de acordo com a mudança da média móvel e não mais de acordo com o valor bruto lido pelo LDR.

O conjunto da média móvel é criado dinamicamente conforme é inserido um valor da leitura do LDR e funciona como uma fila simples onde ao chegar no limite do conjunto, o primeiro elemento que foi inserido é o primeiro elemento a ser retirado, calculando-se a média móvel sempre com os elementos mais recentes.

Para utilizar a média móvel foi importada a biblioteca *movingAvg* no cliente. Seu uso consiste em criar um objeto do tipo *movingAvg* junto com seu tamanho máximo, neste caso foi utilizado o tamanho de 6 elementos e, ao inserir um novo valor no conjunto, a função retorna o cálculo da média móvel atual. Dessa forma, o sistema ficará responsivo à média móvel, então para que o sistema realize uma mudança é necessário que haja uma alteração nas últimas seis leituras do LDR.

Devido às diferentes leituras de luminosidade em diversos ambientes, foi necessário incluir um processo de calibração do sensor ao inicializar o sistema. Assim, a calibração é executada durante os primeiros 15 segundos após o sistema ser iniciado. Durante este tempo, pede-se ao usuário simular a máxima incidência de luz e a ausência dela próximo ao LDR, conforme o função implementada na Figura 3.

```

1 void calibration() {
2   Serial.println(">Starting_calibration...");
3   Serial.println(">Do_this_calibration_in_the_dark_and_simulate_extreme_conditions_of_light");
4
5   //executa a calibracao por n segundos (15)
6   while (millis() < seconds_calibration * 1e3) {
7
8     sensor_ldr = analogRead(A0);
9
10    if (sensor_ldr > sensor_max) { //armazena o valor maximo lido pelo sensor
11      sensor_max = sensor_ldr;
12    }
13    if (sensor_ldr < sensor_min) { //armazena o valor minimo lido pelo sensor
14      sensor_min = sensor_ldr;
15    }
16    delay(50);
17  }
18
19  threshold = (sensor_max + sensor_min) / 2;
20
21  Serial.println(">Calibration_completed!");
22  Serial.print(">Calibration_sensor_MIN_value:"); Serial.println(sensor_min);
23  Serial.print(">Calibration_sensor_MAX_value:"); Serial.println(sensor_max);
24  Serial.print(">Calculated_light_threshold:"); Serial.println(threshold);
25 }

```

Figura 3. Algoritmo 1: Função responsável por realizar a calibração do sistema.

Por fim, na função de calibração é realizado o cálculo do *threshold*, como mostrado na equação (1), dado pela média dos valores máximo e mínimo obtidos anteriormente.

$$threshold = \frac{(sensor\_max + sensor\_min)}{2} \quad (1)$$

Assim, quando houver uma incidência de luz menor que o *threshold* o cliente informará o servidor para acionar os LEDs automaticamente, através do envio de uma variável de controle denominada *ledControl*. Essa comunicação entre cliente e servidor é feita através requisições HTTP, com o uso da biblioteca *Ethernet*.

### 3.2 Sistema Automático da Temperatura de Cor

O sistema automático da temperatura de cor trata-se de uma automação mais intuitiva e mais completa quando comparada com a maioria dos aplicativos atuais disponíveis no mercado. Ao acessar a aplicação *web*, o usuário se depara com a tela inicial mostrada na Figura 4, na qual é possível inserir informações de localização, hora de dormir e definir a temperatura de cor para três momentos do dia, além de poder visualizar graficamente a variação da temperatura de cor ao longo do dia de acordo com os horários e valores de temperatura definidos, o que facilita o entendimento das transições ao longo das 24 horas.

A localização serve para obter a hora em que o sol se põe no local especificado e realizar a transição da temperatura de cor para uma cor mais quente de preferência. E o horário

em que o usuário costuma ir dormir serve para dimerizar o LED até seu desligamento total. A aplicação depende desses dois momentos do dia, pôr do sol e hora de dormir, para realizar a automação e possui valores pré-definidos de pôr do sol às 17h00 e horário de dormir às 22h30. A inserção de tais informações pelo usuário melhora a experiência do próprio usuário uma vez que a automação será baseada no ambiente e no comportamento do usuário.

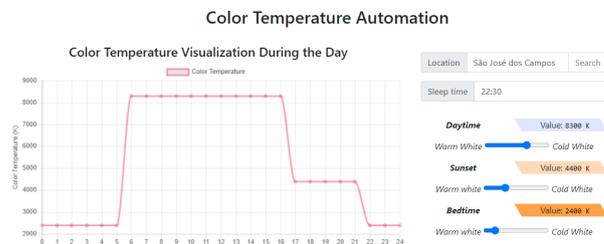


Figura 4. Captura de tela da aplicação *web* da definição de temperatura de cor ao longo do dia.

Para definir as temperaturas de cor para os três momentos do dia, a aplicação possui três entradas em formato de barras deslizantes que variam de 1000K, correspondente ao branco quente, até 12000K, correspondente ao branco frio. Os 3 momentos podem ser definidos como:

- (1) *Daytime*: fase do dia em que o usuário estará acordado, durante o horário comercial;
- (2) *Sunset*: fase do dia em que o usuário estará acordado, no momento após o pôr do sol, em que a luminosidade ambiente começara a cair naturalmente;
- (3) *Bedtime*: período em que o usuário estará dormindo, teoricamente não havendo uso das luzes.

Os valores das temperaturas de cor foram obtidos de Charity (2001) e utilizados como um dicionário, onde cada temperatura de cor corresponde a um valor no formato RGB.

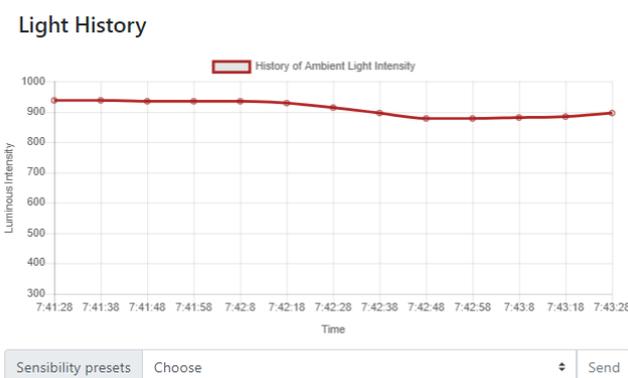


Figura 5. Captura de tela da aplicação *web* do histórico de intensidade luminosa ambiente.

A segunda parte da aplicação *web* consiste em mostrar o histórico dos valores da média móvel enviados pelo cliente, conforme exibido na Figura 5. Além do mais, o usuário pode escolher a sensibilidade do sistema em resposta à mudança de luminosidade no campo *Sensibility presets*, onde há três modos de atualização dos dados:

- (1) *Fast*: rápida taxa de atualização do sistema (a cada 10 segundos). Acaba se tornando desvantajosa em ambientes onde há muitas variações de luz, levando à um consumo maior de energia pelo sistema;
- (2) *Recommended*: ocorre uma coleta e uma atualização do sistema a cada dois minutos e meio (150 segundos) e é a opção recomendada;
- (3) *Slow*: uma taxa de atualização mais lenta (a cada 5 minutos ou 300 segundos), o que implica em um menor consumo de energia por parte do sistema.

Como o conjunto de valores da média móvel possui limite de 6 elementos, o sistema demora 6 ciclos de coleta de dados para ter uma reação de acionamento do LED, então, para uma aquisição de dados a cada 150 segundos, por exemplo, o tempo de resposta à luz seria de 15 minutos (900 segundos).

### 3.3 Fluxo de dados

Ao acessar a aplicação, o usuário é capaz de inserir dados de localização, a hora que ele costuma dormir e as três temperaturas de cor desejadas para cada momento do dia. Com o conhecimento dessas informações, a aplicação é capaz de criar o gráfico de temperatura de cor ao longo do dia, auxiliando o usuário a compreender o funcionamento do sistema.

Além disso, o usuário também pode visualizar a variação da intensidade luminosa em tempo real e definir a sensibilidade de captura dos dados dentre as três opções de *presets* disponíveis: uma coleta a cada 10 segundos, uma a cada 150 segundos ou uma a cada 300 segundos.

Todas as funcionalidades da aplicação *web* descritas acima podem ser vistas no diagrama de caso de uso da aplicação na Figura 6.

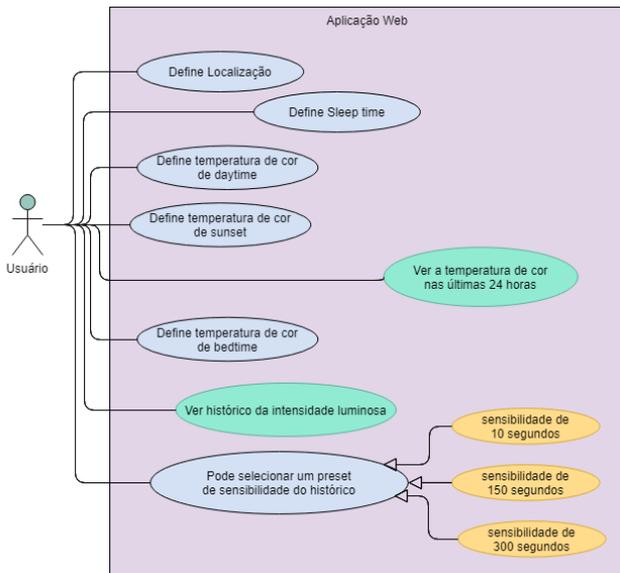


Figura 6. Diagrama de casos de uso entre o usuário e aplicação *web*.

Já na Figura 7 é possível visualizar o funcionamento do cliente, do servidor e da aplicação *web* e o relacionamento entre eles através do diagrama de atividades do sistema.

objetos oriundos de bibliotecas, como *movingAvg* (cálculo da média móvel), *Ticker* (repetição de função em um intervalo de tempo) e *ESP8266WebServer* (criação das instâncias do cliente e do servidor).

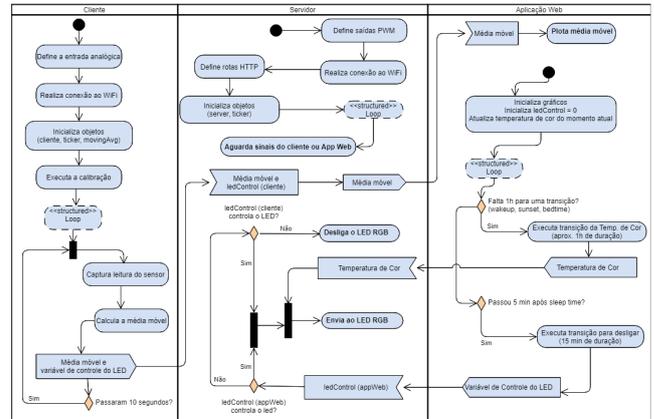


Figura 7. Diagrama de atividade do sistema completo.

No subsistema do cliente, há a definição da porta analógica onde será feita a leitura do LDR. Em seguida, é realizada a conexão em um ponto de acesso sem fio (*Wireless Access Point*, WAC ou AC) com o uso da biblioteca *WiFi*, e inicializado seus objetos (cliente, *ticker* e *movingAvg*). No final da função *setup*, é chamada a função de calibração, que passa a ser executada nos primeiros 15 segundos para realizar o cálculo do *threshold*. Por fim, na função *loop* do cliente é realizada uma aquisição do valor do LDR a cada 10 segundos, calculado a média móvel, e enviado a média móvel junto com a variável de controle ao servidor.

Já no subsistema do servidor, é definido as portas de saída com PWM onde será conectado o LED RGB. Em seguida é realizada a conexão ao AC e definida as rotas de requisição HTTP. Por fim, na função *loop* a tarefa do servidor é repassar a média móvel à aplicação *web* e aguardar os sinais oriundos do cliente e da aplicação para realizar o controle do LED RGB.

O valor da média móvel da iluminação ambiente é consumido pela aplicação *web* e plotado em um gráfico para que seja de conhecimento do usuário. Assim, a inicialização da aplicação *web* é dada pela criação dos gráficos de Temperatura de Cor e do histórico de intensidade luminosa, e inicializa a variável de controle do LED RGB passando o controle para o cliente (realizar o acionamento do LED), e informa ao servidor a Temperatura de Cor do momento atual, assim caso o acionamento do LED seja feito, o mesmo se acenderá na Temperatura de Cor do momento correto.

A função *loop* da aplicação consiste em verificar a cada minuto duas condições: se falta exatamente uma hora para realizar uma transição de temperatura de cor, em caso positivo inicia-se a transição com duração aproximada de uma hora para a nova Temperatura de Cor, e a segunda verificação é caso passou-se cinco minutos após o *sleep time*, pois o sistema iria realizar o desligamento gradativo dos LEDs com uma duração aproximada de 15 minutos.

#### 4. IMPLEMENTAÇÃO EXPERIMENTAL E RESULTADOS

Para comprovar o funcionamento da lâmpada inteligente simulada e de sua automação, foram realizados testes de bancada para testar o acionamento automático do LED e sua dimerização. Por último, foram testadas as diversas Temperaturas de Cor e suas transições ao longo do dia.

##### 4.1 Acionamento do LED e Dimerização Automática

No teste apresentado nas Figuras 8 e 9, foi utilizado a taxa de aquisição de dados no modo rápido (uma coleta a cada 10 segundos). Sendo assim, iniciou-se com o ambiente em alta claridade, em seguida foi-se diminuindo a luminosidade ambiente gradativamente até o momento em que o LED RGB acendesse automaticamente pela luminosidade ficar abaixo do *threshold*.



Figura 8. Histórico da luminosidade do ambiente durante o teste.

Como é possível verificar na Figura 9 (a), o LED manteve-se desligado enquanto a luminosidade ambiente estava acima do *threshold*, pois o sistema considera a iluminação adequada. Entretanto, conforme a luminosidade do ambiente decaía, a intensidade luminosa do LED aumentava proporcionalmente conforme mostrado na Figura 9 (b) em diante.

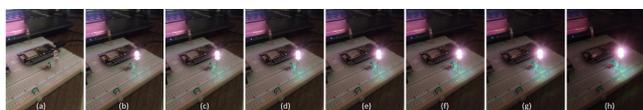


Figura 9. Fotos tiradas do LED RGB conforme variava-se a luz ambiente.

Na Tabela 1 é possível verificar a intensidade luminosa percentual de cada caso da Figura 9, podendo ser comparada ao *duty cycle* equivalente, pois o controle da intensidade do LED foi realizado por PWM.

Tabela 1. Ativação e dimerização automática do LED RGB.

Foto	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
Intensidade	off	10%	14%	20%	24%	33%	49%	100%

##### 4.2 Históricos da Intensidade Luminosa

A seguir serão mostradas algumas situações em que foram registrados os históricos da média móvel da luminosidade do ambiente.

No primeiro teste, apresentado na Figura 10, houve uma variação da intensidade luminosa no período da manhã, na qual foi aberta a cortina do ambiente às 7 horas e 34 minutos e aproximadamente 30 segundos. É possível verificar claramente o funcionamento do filtro de média móvel pois houve uma mudança abrupta da claridade no local, porém o sinal foi suavizado e demorou exatamente seis ciclos de coleta de dados para estabilizar.

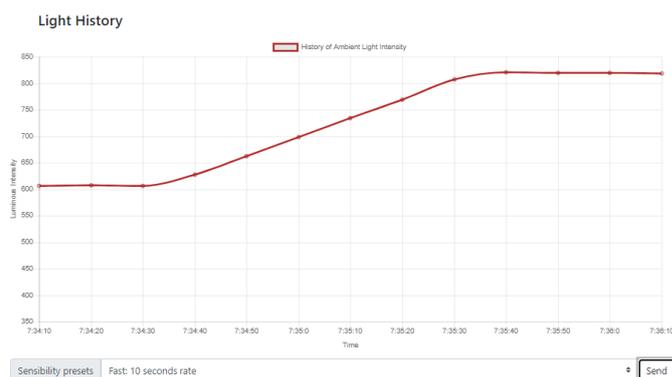


Figura 10. Histórico da intensidade luminosa no período da manhã com abertura de cortina.

O teste da Figura 11 foi realizado ao meio-dia. Nele observa-se uma iluminação do ambiente constante até o horário de 12 horas 42 minutos e 51 segundos, a partir desse momento iniciou-se um período de chuva fraca, o suficiente para conseguir diminuir gradativamente a leitura do sensor a partir das 12 horas e 48 minutos.

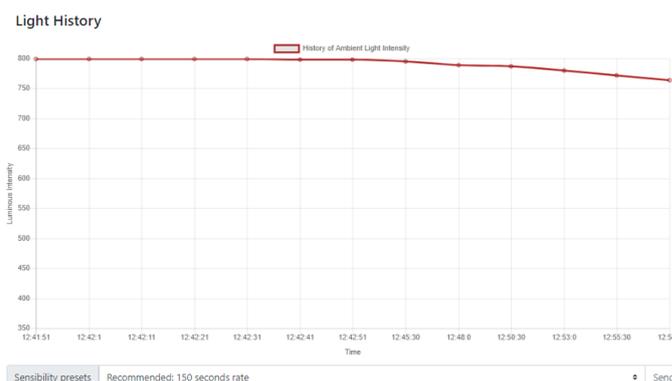


Figura 11. Histórico da intensidade luminosa ao meio-dia com início de uma chuva.

No teste exibido na Figura 12, teve uma hora de duração e ocorreu no período do final da tarde, antecedendo o pôr do sol. Nele é possível verificar o decaimento natural da intensidade luminosa. Além do mais, no ponto destacado (no instante 18 horas 9 minutos e 38 segundos) houve o acionamento automático da *smart* lâmpada simulada, pois a intensidade luminosa ambiente excedeu o valor do *threshold*.

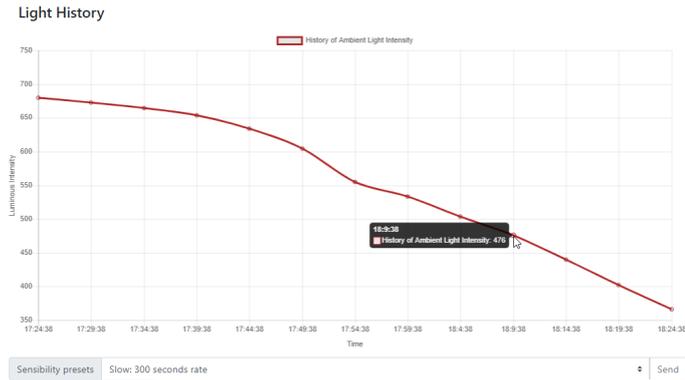


Figura 12. Histórico da intensidade luminosa no período da tarde, precedendo o pôr do sol.

### 4.3 Transição da Temperatura de Cor

No teste mostrado a seguir, foi realizada a maior transição de Temperatura de Cor: variando de 12000 K a 1000 K. Porém, como pequenas variações na Temperatura de Cor não são tão perceptíveis, a transição realizada no teste variou de acordo com os valores da Tabela 2.

Tabela 2. Tabela com as temperaturas de cor (em Kelvin) realizada no teste.

(a) 12000	(b) 11000	(c) 10000	(d) 9000
(e) 8000	(f) 7000	(g) 6500	(h) 6000
(i) 5500	(j) 5000	(k) 4500	(l) 4000
(m) 3600	(n) 3200	(o) 3100	(p) 3000
(q) 2900	(r) 2800	(s) 2700	(t) 2600
(u) 2500	(v) 2400	(w) 2300	(x) 2200
(y) 2100	(z) 2000	(aa) 1900	(bb) 1800
(cc) 1700	(dd) 1600	(ee) 1500	(ff) 1400
(gg) 1300	(hh) 1200	(ii) 1100	(jj) 1000

Na Figura 13 há uma sequência de capturas fotográficas obtidas da lâmpada inteligente simulada nas Temperaturas de Cor descritas na Tabela 2. Entretanto, para uma melhor percepção das diferentes Temperaturas de Cores, as fotografias foram tiradas em um ambiente escuro, como também foi utilizado um difusor posicionado em cima do LED RGB.



Figura 13. Variação da temperatura de cor.

Para comprovar as transições de temperatura de cor, foi configurada a automação descrita na Figura 14. Nela, ocorrerá duas transições de temperatura de cor e uma transição da dimerização do LED RGB até seu desligamento total. A primeira transição inicia-se às 17h com a Temperatura de Cor de *daytime* (4600 K) e acaba às 18h

com a Temperatura de Cor de *sunset* (3000 K). A segunda transição inicia-se às 18h em 3000 K e finaliza às 19h com a Temperatura de Cor de *bedtime* (1600 K). Por fim, ao chegar no momento de *sleep time* (19h30) com acréscimo de 5 minutos, inicia-se a dimerização gradual da *smart* lâmpada simulada até o seu desligamento total em 19h45.

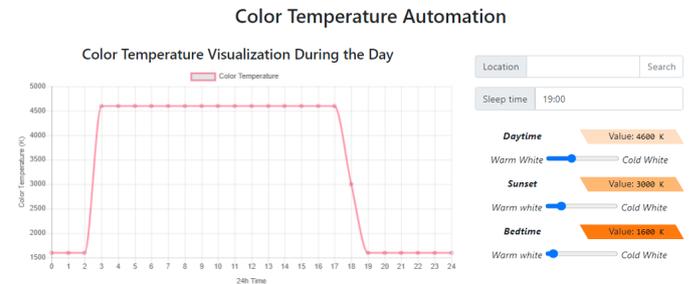


Figura 14. Captura de tela contendo a aplicação *web* e a configuração da temperatura de cor.

Na Figura 15 é mostrado o registro da comunicação entre a aplicação e o servidor. Como a verificação das automações é realizada na aplicação, é necessário comunicar ao servidor utilizando requisições HTTP, enviando os comandos das mudanças de Temperatura de Cor no período de aproximadamente uma hora. Nele é possível verificar a primeira transição da Temperatura de Cor de *daytime* para *sunset*, saindo de 4600 K chegando a 3000 K.

```

Sunset time! Starting transition to sunset_kelvin
Starting transition to warmer white!
Sending transition color: 255, 219, 186
Sending transition color: 255, 217, 182
Sending transition color: 255, 215, 177
Sending transition color: 255, 213, 173
Sending transition color: 255, 211, 168
Sending transition color: 255, 209, 163
Sending transition color: 255, 206, 159
Sending transition color: 255, 204, 153
Sending transition color: 255, 201, 148
Sending transition color: 255, 199, 143
Sending transition color: 255, 196, 137
Sending transition color: 255, 193, 132
Sending transition color: 255, 190, 126
Sending transition color: 255, 187, 120
Sending transition color: 255, 184, 114
Sending transition color: 255, 180, 107
    
```

Figura 15. Transição da temperatura de cor de *daytime* (4600 K) para *sunset* (3000 K).

Assim como a primeira transição, na Figura 16 há o registro da comunicação entre a aplicação e o servidor durante a transição da Temperatura de Cor de *sunset* (3000 K) para *bedtime* (1600 K), com duração aproximada de uma hora.

```
Sleep time! Starting transition to bedtime_kelvin
Starting transition to warmer white!
Sending transition color: 255, 177, 101
Sending transition color: 255, 173, 94
Sending transition color: 255, 169, 87
Sending transition color: 255, 165, 79
Sending transition color: 255, 161, 72
Sending transition color: 255, 157, 63
Sending transition color: 255, 152, 54
Sending transition color: 255, 147, 44
Sending transition color: 255, 142, 33
Sending transition color: 255, 138, 18
Sending transition color: 255, 131, 0
Sending transition color: 255, 126, 0
Sending transition color: 255, 121, 0
Sending transition color: 255, 115, 0
```

Figura 16. Transição da temperatura de cor de *sunset* (3000 K) para *bedtime* (1600 K).

Por fim, na Figura 17 está o registro de comunicação entre a aplicação *web* e o servidor da transição de dimerização do LED RGB. Os valores de 1 a 10 é interpretado pelo servidor e são inversamente proporcionais ao *duty cycle* enviado ao LED RGB, sendo que o valor 11 corresponderia ao desligamento total da *smart* lâmpada simulada.

```
Turning off time (sleep_time + 5 min)! Turning off the lights
Sending transition Off: 1
Sending transition Off: 2
Sending transition Off: 3
Sending transition Off: 4
Sending transition Off: 5
Sending transition Off: 6
Sending transition Off: 7
Sending transition Off: 8
Sending transition Off: 9
Sending transition Off: 10
Sending transition Off: 11
```

Figura 17. Transição do desligamento do LED devido o horário de *sleep time*.

## 5. CONCLUSÃO

Este artigo apresentou um sistema automático de iluminação que atua de forma mais intuitiva e simplificada para o usuário do que as lâmpadas inteligentes disponíveis no mercado. A lâmpada inteligente, representada por um LED RGB conectado a um servidor, comporta-se conforme as informações recebidas do cliente e da aplicação *web*. Com o sistema desenvolvido é possível que o usuário defina a temperatura de cor desejada ao longo do dia, e que as mudanças ocorram de forma suave, gradualmente. Além disso, este sistema possui um controle da iluminação de acordo com a luminosidade do ambiente. Com os resultados obtidos em testes experimentais, pôde-se ver que o sistema proporcionou o controle automático da intensidade luminosa e da temperatura de cor para o usuário, aproveitando melhor os recursos das *smart* lâmpadas e facilitando seu uso. Adicionalmente, o uso consciente da temperatura de cor resulta na regulação dos raios de luz, evitando a luz azul no período que precede o sono, proporcionando um melhor conforto e qualidade de vida ao usuário.

## REFERÊNCIAS

- Almeida, R.M.A., de Moraes, C.H.V., and Seraphim, T.d.F.P. (2017). *Programação de Sistemas Embarcados: Desenvolvendo Software para Microcontroladores em Linguagem C*. Elsevier Brasil.
- Arecchi, A.V., Messadi, T., and Koshel, R.J. (2007). *Field Guide to Illumination*. Society of Photo-Optical Instrumentation Engineers (SPIE), USA.
- Buso, S. and Spiazzi, G. (2011). White light solid state lamp with luminance and color temperature control. In *XI Brazilian Power Electronics Conference*, 837–843. doi:10.1109/COBEP.2011.6085167.
- Byun, J., Hong, I., Lee, B., and Park, S. (2013). Intelligent household led lighting system considering energy efficiency and user satisfaction. *IEEE Transactions on Consumer Electronics*, 59(1), 70–76. doi:10.1109/TCE.2013.6490243.
- Charity, M. (2001). Blackbody color datafile. URL [http://www.vendian.org/mncharity/dir3/blackbody/UnstableURLs/bbr\\_color.html](http://www.vendian.org/mncharity/dir3/blackbody/UnstableURLs/bbr_color.html).
- Chellappa, S.L., Steiner, R., Blattner, P., Oelhafen, P., Götz, T., and Cajochen, C. (2011). Non-visual effects of light on melatonin, alertness and cognitive performance: Can blue-enriched light keep us alert? *PLOS ONE*, 6(1), 1–11. doi:10.1371/journal.pone.0016429.
- Kim, H.S. and Lee, Y.H. (2019). Correlation analysis of image reproduction and display color temperature change to prevent sleep disorder. *IEEE Access*, 7, 59091–59099. doi:10.1109/ACCESS.2019.2914768.
- Kulve, M., Schlangen, L., Schellen, L., Souman, J., and Lichtenbelt, W. (2018). Correlated colour temperature of morning light influences alertness and body temperature. *Physiology & Behavior*, 185, 1–13.
- Kurose, J., Ross, K., Marques, A., and Zucchi, W. (2007). *Redes de computadores e a Internet: uma abordagem top-down*. Pearson Addison Wesley.
- Lee, A.T.L., Chen, H., Tan, S.C., and Hui, S.Y. (2016). Precise dimming and color control of led systems based on color mixing. *IEEE Transactions on Power Electronics*, 31(1), 65–80. doi:10.1109/TPEL.2015.2448641.
- Muhamad, W.N.W., Mohd Anuar, A.S., Sarnin, S.S., Mohamad Azreen, M., Idris, A., and Kassim, M. (2018). Intelligence lighting control system with precise dimming and correlated color temperature. In *TENCON 2018 - 2018 IEEE Region 10 Conference*, 1235–1240. doi:10.1109/TENCON.2018.8650228.
- Statista (2022). Statista comfort and lighting - worldwide. <https://www.statista.com/outlook/dmo/smart-home/comfort-lighting/worldwide>. Acessado em 15 de Agosto de 2022.
- Viola, A.U., James, L.M., Schlangen, L.J., and Dijk, D.J. (2008). Blue-enriched white light in the workplace improves self-reported alertness, performance and sleep quality. *Scandinavian Journal of Work, Environment & Health*, 34(4), 297–306.
- Ákos, B., Árpád, S., and János, S. (2001). The concept of correlated colour temperature revisited. *Color Research & Application*, 26(6), 450–457. doi:https://doi.org/10.1002/col.1065.