

---

---

# **<LilianProject> Documento de Arquitetura de Software**

**Versão 0.3**

<Nome do Projeto>	Version: <1.0>
Documento de Arquitetura de Software	Date: <dd/mmm/yy>
<document identifier>	

## Histórico da Revisão

Data	Versão	Descrição	Autor
02/11/2018	0.1	Criação do documento	Roberto Kenzo Hamano
02/11/2018	0.2	Preenchimento de informações	Roberto Kenzo Hamano
02/11/2018	0.3	Preenchimento de informações	Djair Junio Mendes Batista

<Nome do Projeto>	Version: <1.0>
Documento de Arquitetura de Software	Date: <dd/mmm/yy>
<document identifier>	

## Índice Analítico

1.	Introdução	4
1.1	Finalidade	4
1.2	Escopo	4
1.3	Definições, Acrônimos e Abreviações	4
1.4	Referências	4
1.5	Visão Geral	4
2.	Representação Arquitetural	4
3.	Metas e Restrições da Arquitetura	4
4.	Visão de Casos de Uso	5
4.1	Realizações de Casos de Uso	5
5.	Visão Lógica	5
5.1	Visão Geral	5
5.2	Pacotes de Design Significativos do Ponto de Vista da Arquitetura	5
6.	Visão de Processos	5
7.	Visão de Implantação	5
8.	Visão da Implementação	5
8.1	Visão Geral	5
8.2	Camadas	6
9.	Visão de Dados (opcional)	6
10.	Tamanho e Desempenho	6
11.	Qualidade	6

<Nome do Projeto>	Version: <1.0>
Documento de Arquitetura de Software	Date: <dd/mmm/yy>
<document identifier>	

# Documento de Arquitetura de Software

## 1. Introdução

### 1.1 Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema. O documento abrange mais especificamente a arquitetura do projeto em si. O público-alvo do documento é o arquiteto de software e os outros membros da equipe do projeto, sendo um meio de comunicação entre ambos.

### 1.2 Escopo

O Documento de Arquitetura de Software tem como objetivo fornecer uma visão geral da arquitetura do sistema de software, sendo o meio de comunicação entre o arquiteto de software e os demais membros da equipe do projeto, com relação a decisões decisivas sob o ponto de vista da arquitetura.

### 1.3 Definições, Acrônimos e Abreviações

Lilian, acrônimo para “Light Linear Assistant for Networking”.

### 1.4 Referências

API do Telegram: <http://github.com/eternnoir/pytelegrambotapi>

### 1.5 Visão Geral

O restante do Documento de Arquitetura de Software contém visões sob diferentes aspectos do projeto assim como os requisitos não funcionais do sistema.

## 2. Representação Arquitetural

A arquitetura tem restrições apenas do framework pyTelegramBotAPI.

## 3. Metas e Restrições da Arquitetura

O software tem a arquitetura restrita à arquitetura de construção do framework pyTelegramBotAPI.

### Visão de Casos de Uso

- O usuário geral pode cadastrar, editar, gerenciar canais
- O usuário geral pode criar, editar, gerenciar postagens
- O usuário geral pode publicar propagandas em seu nome ou em benefício
- O administrador de canais pode manter as informações de processos
- O administrador de canais pode gerar relatórios gerenciais
- O administrador de canais pode consultar processos

### 3.1 Realizações de Casos de Uso

*[Esta seção ilustra o funcionamento do software, apresentando algumas realizações (ou cenários) de casos de uso selecionadas e explica como os diversos elementos do modelo de design contribuem para a respectiva funcionalidade.]*

<Nome do Projeto>	Version: <1.0>
Documento de Arquitetura de Software	Date: <dd/mm/yy>
<document identifier>	

## Visão Lógica

### 3.2 Visão Geral

*[Esta subseção descreve toda a decomposição do modelo de design em termos de camadas e de hierarquia de pacotes.]*

### 3.3 Pacotes de Design Significativos do Ponto de Vista da Arquitetura

*[Para cada pacote significativo, inclua uma subseção com o respectivo nome, uma breve descrição e um diagrama com todos os pacotes e classes significativos nele contidos.*

*Para cada classe significativa no pacote, inclua o respectivo nome, uma breve descrição e, opcionalmente, uma descrição de algumas das suas principais responsabilidades, operações e atributos.]*

## 4. Visão de Processos

*[Esta seção descreve a decomposição do sistema em processos leves (threads simples de controle) e processos pesados (agrupamentos de processos leves). Organize a seção em grupos de processos que se comunicam ou interagem. Descreva os modos principais de comunicação entre processos, como transmissão de mensagens e interrupções.]*

## 5. Visão de Implantação

*[Esta seção descreve uma ou mais configurações da rede física (hardware) na qual o software é implantado e executado. Ela é uma visão do Modelo de Implantação. No mínimo, para cada configuração, ela deve indicar os nós físicos (computadores, CPUs) que executam o software e suas interconexões (barramento, LAN, ponto a ponto, etc.) É incluído também um mapeamento dos processos da **Visão de Processos** nos nós físicos.]*

## Visão da Implementação

### 5.1 Visão Geral

*[Esta subseção nomeia e define as diversas camadas e o seu conteúdo, as regras que determinam a inclusão em uma camada específica e as fronteiras entre as camadas. Inclua um diagrama de componentes que mostre os relacionamentos entre as camadas. ]*

### 5.2 Camadas

*[Para cada camada, inclua uma subseção com o respectivo nome, uma lista dos subsistemas localizados na camada e um diagrama de componentes.]*

## 6. Tamanho e Desempenho

O software possui um tamanho de 8mb e um desempenho de respostas em no máximo 70ms.

## 7. Qualidade

A arquitetura do software possui uma baixa qualidade, portanto não contribui para todos os recursos do sistema.