AP2

Documento de Arquitetura de Software

Versão <1.0>

Disciplina

Programação Avançada

Professor

Victor Hazin

Equipe

Allyson Halley 200920338

João Nascimento 171081856

Noé Neto 152080449

Victor Viana 200919838

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 22/09/2018 | Alpha 1 | Início de estrutura | Victor Viana |
| 18/10/2018 | Alpha 2 | Backlog Trello | Victor Viana |
| 18/10/2018 | 0.1 | Implementação Android | Welber |
| 19/10/2018 | 0.2 | Estruturação Android | Allyson Halley |
| 15/11/2018 | 0.2 | Player | Noé Oliveira |
| 15/11/2018 | 0.3 | Início da documentação | Allyson Halley |
| 26/11/2018 | 1.0 | Play Music | Noé Oliveira |
| 27/11/2018 | 1.1 | Play Video | Noé Oliveira |
| 28/11/2018 | 1.1.b | Revisão de Código | Grupo |
| 28/11/2018 | 1.2 | Estruturação da documentação | Allyson Halley |

Índice

1. Introdução 4

1.1 Objetivo 4

1.2 Escopo 4

1.3 Referências 4

2. Representação Arquitetural 4

3. Restrições e Metas Arquiteturais 4

3.1 Restrições 4

3.2 Metas Arquiteturais 4

4. Visão de Casos de Uso 4

5. Visão Lógica 4

5.1 Visão Geral 4

5.2 Camadas 4

6. Visão de Dados 5

Documento de Arquitetura de Software

# Introdução

Se trata de um projeto acadêmico com fins apenas acadêmicos. Em suma montar um gerenciador de downloads mostrando os pontos requisitados em documentação entregue pelo professor.

## Objetivo

Resumir em um documento simples os pontos requisitados pelo professor da disciplina.

## Escopo

Este documento vai guiar o professor nas limitações e alcance nos pontos requisitados.

## Referências

Documentos de requisitos do projeto:  
https://docs.google.com/document/d/1jTiS2i68D9C\_pQOGBi1b-ZfnDS118yWN6S6dgF6rs\_4/edit

# Representação Arquitetural

[Esta seção descreve qual é a arquitetura de software do sistema atual e como ela é representada. Dos **Casos de Uso**, **Implementação**, **Processo** , **Lógica**e **Visualizações de Implementação**, ela enumera as visualizações necessárias e, para cada uma, explica que tipos de elementos de modelos a mesma contém.]

# Restrições e Metas Arquiteturais

## Restrições

O sistema não vai ultrapassar os requisitos descritos no documento referência.

## Metas Arquiteturais

Deixar as macro-funções de cada classe restrita a seu grupo funcional respeitando a ideia principal do MVC e da arquitetura proposta pelo Android.

# Visão de Casos de Uso

Vide proposta do projeto da disciplina.

# Visão Lógica

Na arquitetura temos uma Lib provendo conexões, e as comunicações externas ao Android.

## Visão Geral

O software tem como fim gerenciar downloads em threads paralelas com fila de execução, garantindo a execução de todos na fila de espera de downloads.

## Camadas

Manifest: Instruções básicas de tratamento generalista das telas do “res”;

MVC: onde encontra-se a pasta “Java” com toda a regra de negócio.

Modelo (M): Classes bases, objetos;

View(V): Classes que se ligam e dão funções as telas e seus componentes;

Controller (C): Representados pelas classes DAO, onde estão as conexões de persistências ou similares.

RES: dentro da estrutura promovida pelo Android, nesta pasta se encontra toda modelagem frontend do app através de manipulação de XML;

Lib: Nela estão as comunicações externas ao Android.

# Padrões Utilizados

## Singletone

Está sendo utilizada em “DownloadManager.java”, na raiz da lib, para garantir acesso por uma única instância de toda a aplicação.

## Strategy

## Command

Também incluído na classe “DownloadManagerRequest.java” para abstrair as funções que serão chamadas a medida que o download for finalizado.

## Builder

Incluído em “DownloadManagerRequest.java” para facilitar a utilização da lib.

## Adapter

# Visão de Dados

Dados diferentes dos arquivos manipulados pelos gerenciador de downloads serão persistidos localmente, similar a cache.