 **INSTITUTO POLITÉCNICO DE BEJA **

**Escola Superior de Tecnologia e Gestão**

**Tecnologias Web e Dispositivos Móveis**

**Programação de Aplicação do Lado do Cliente**

**Trabalho de Grupo #1**

**Catálogo Eletrónico de Livros da Biblioteca da Escola Santiago Maior**

**Elaborado por:**

**Tiago Miguel Nunes Caracol**

**João Daniel Candeias Costa**

Beja

17/06/2021

 **INSTITUTO POLITÉCNICO DE BEJA **

**Escola Superior de Tecnologia e Gestão**

**Tecnologias Web e Dispositivos Móveis**

**Programação de Aplicação do Lado do Cliente**

**Trabalho de Trabalho de Grupo #1**

**Catálogo Eletrónico de Livros da Biblioteca da Escola Santiago Maior**

**Elaborado por:**

**Tiago Miguel Nunes Caracol**

**João Daniel Candeias Costa**

Docente:

Luís Rosário, David Fontes, Carlos Sancho

Beja

17/06/2021

Resumo

Este relatório, elaborado no âmbito do Curso de Tecnologias Web e Dispositivos Móveis expõe as nossas competências adquiridas ao longo deste semestre nas disciplinas de Programação para Dispositivos Móveis e Sistemas Interativos

A escolha deste projeto:

Catálogo Eletrónico de Livros da Biblioteca da Escola Santiago Maior – Utilizamos o mesmo tema de trabalho do semestre passado.

Abstract

This report, prepared in the context of the Web Technologies and Mobile Devices Course, exposes our skills acquired during this semester in the disciplines of Programming for Mobile Devices and Interactive Systems.

The choice of the project:

Catálogo Eletrónico de Livros da Biblioteca da Escola Santiago Maior – We used the same theme as last semester.

Índice Geral

[1 Introdução 1](#_Toc74873823)

[2 Organização do trabalho e ferramentas 2](#_Toc74873824)

[3 Análise do problema 3](#_Toc74873825)

[4 Funcionalidades da aplicação 4](#_Toc74873826)

[5 Interface da aplicação 6](#_Toc74873827)

[5 Interface da aplicação 7](#_Toc74873828)

[5 Interface da aplicação 8](#_Toc74873829)

[5 Interface da aplicação 9](#_Toc74873830)

[5 Interface da aplicação 10](#_Toc74873831)

[5 Interface da aplicação 11](#_Toc74873832)

[5 Interface da aplicação 12](#_Toc74873833)

[5 Interface da aplicação 13](#_Toc74873834)

[5 Interface da aplicação 14](#_Toc74873835)

[6 Testes com a aplicação 15](#_Toc74873836)

[7 Partes importantes do código 16](#_Toc74873837)

[7 Partes importantes do código 17](#_Toc74873838)

[7 Partes importantes do código 19](#_Toc74873839)

[8 Conclusão 21](#_Toc74873840)

[9 Referências Bibliográficas 22](#_Toc74873841)

Índice de Figuras

[Figura 1- Smartphone 1](#_Toc74874166)

[Figura 2- Tarefas 2](#_Toc74874167)

[Figura 3- Homem a pensar 3](file:///C:\Users\Tiago%20Caracol\Desktop\palcword.docx#_Toc74874168)

[Figura 4- menu pricipal 6](#_Toc74874169)

[Figura 5-menu procurar 1/2 7](#_Toc74874170)

[Figura 6-menu categorias 8](#_Toc74874171)

[Figura 7- Lista arte 9](#_Toc74874172)

[Figura 8-Livro detalhes 10](#_Toc74874173)

[Figura 9-menu favoritos 11](#_Toc74874174)

[Figura 10- menu historico 12](#_Toc74874175)

[Figura 11-menu login 13](#_Toc74874176)

[Figura 12-menu requisitar 14](#_Toc74874177)

[Figura 13- Stickman a certificar tarefas 21](file:///C:\Users\Tiago%20Caracol\Desktop\palcword.docx#_Toc74874178)

# Introdução

O presente projeto pertence à disciplina de PALC (Programação de Aplicações ao lado do cliente do Curso de Tecnologias Web e Dispositivos Móveis do IPBeja. Este tem como principal objetivo abranger todos os conhecimentos adquiridos ao longo deste semestre.

O tema escolhido foi o Catálogo Eletrónico de Livros da Biblioteca da Escola Santiago Maior e este tem como objetivo o desenvolvimento de uma aplicação android para a biblioteca da Escola de Santiago Maior.

A aplicação deverá ser capaz de dar a oportunidade ao utilizador de poder requisitar um livro através da sua aplicação android.

A aplicação contém um menu com várias funcionalidades, desde poder procurar livros por os seus nomes, procurar livros por categorias, adicionar um livro aos favoritos e consequentemente ver essa lista de favoritos e também dispõem de uma lista com todos os livros requisitados pelo utilizador.

Uma imagem com texto

Descrição gerada automaticamente

Figura 1- Smartphone

# Organização do trabalho e ferramentas

Com o tema do trabalho definido, estava na altura de começar em grupo a pensar nos layouts que iriamos utilizar na aplicação.

Em conjunto começamos por fazer pequenos esboços em papel, até futuramente replicarmos no Abode XD. Quando concluímos essa etapa, começamos por replicar os diferentes layouts no android Studio usando a matéria lecionada nas aulas de Sistemas Interativos.

A seguir, tivemos que chegar a uma conclusão de quias os campos que iriamos usar na base de dados do trabalho, concluindo isso, fizemos a construção da base de dados dentro da aplicação do android Studio, usando a biblioteca room, lecionado na disciplina de programação de dispositivos móveis.

Dividimos o resto do trabalho por activitys e cada um de nós ficava encarregue de fazer as suas. A partir daí o nosso trabalho foi mútuo ajudando-nos um ao ouro nas dúvidas que o outro tinha.

Com o projeto quase finalizados, reunimo-nos novamente e em conjunto resolvemos os últimos problemas que achava mos que deviam ser resolvidos.

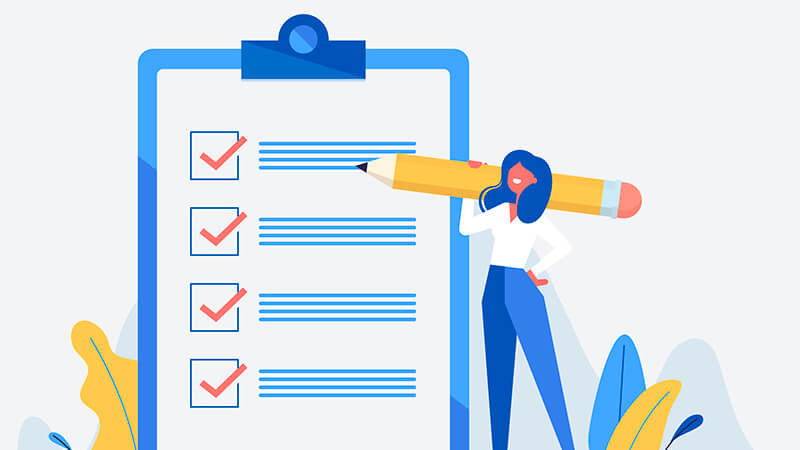


Figura 2- Tarefas

**As ferramentas utilizadas na produção deste trabalho foram:**

Android Studio- Aplicação para desenvolver código e os layouts.

Adobe Ilustrator- Aplicação onde usamos para editar os nossos ícones do menu.

GitHub- Aplicação para descarregar as novas versões.

Adobe XD- Aplicação usada para desenhar os layouts.

# Análise do problema

Naturalmente todas as escolas dispõem uma biblioteca para uso dos seus alunos e professores. Umas das funções da mesma é oferecer um lugar silencioso para os alunos poderem estudar ou mesmo fazer alguns trabalhos necessários nos computadores que a escola disponibiliza nas bibliotecas.

Para os apreciadores de livros, a escola tem uma vasta gama de livros disponíveis para os seus utentes requisitarem. Com o nosso projeto, os alunos só têm de iniciar sessão na aplicação e rapidamente conseguem saber quais os livros disponíveis para eles requisitarem.

Na aplicação os alunos poderão fazer procura de livros que estarão divididos em vários campos, dando assim a oportunidade ao utilizador de poder pesquisar um livro pelo nome do livro, ou por categoria. Poderão também visualizar o seu histórico de livros já requisitados e adicionar livros aos favoritos.

Uma imagem com texto

Descrição gerada automaticamente

Figura 3- Homem a pensar

# Funcionalidades da aplicação

* Consultar as informações de um livro;
* Pesquisa de livros através do nome do livro;
* Pesquisa de livros através de categorias;
* Adicionar e remover livros dos favoritos;
* Requisitar um livro;
* Pesquisa do histórico de livros;

# Interface da aplicação

**Menu intro:**

* Página principal de abertura da aplicação

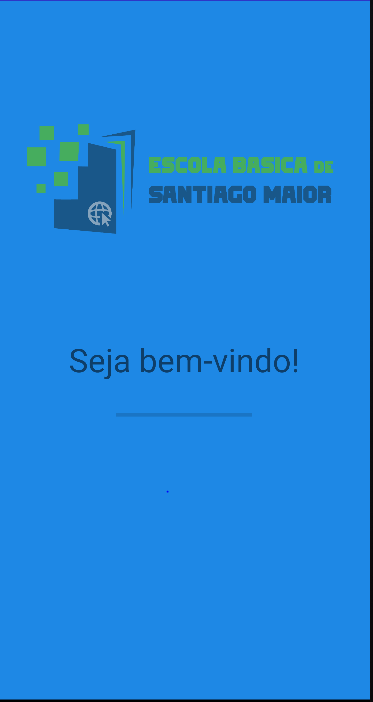


Figura -menu intro

# Interface da aplicação

**Menu principal:**

* Constituído por uma nav bar com ícones, tem uma lista com vários livros, divididos entre populares, romance, arte, etc….
* Dá a possibilidade de o utilizador escolher o livro que deseja visualizar, ou então, navegar nos ícones em baixo para ir até outras páginas.

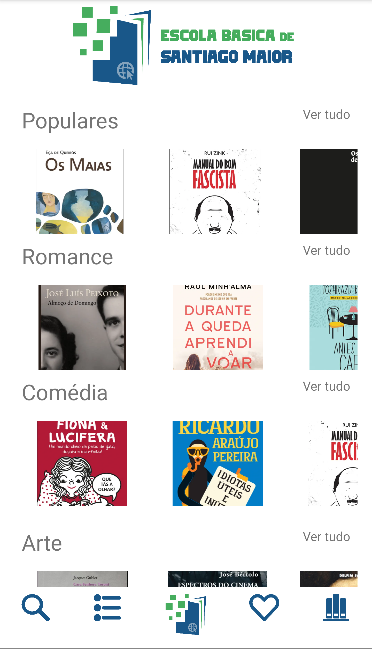
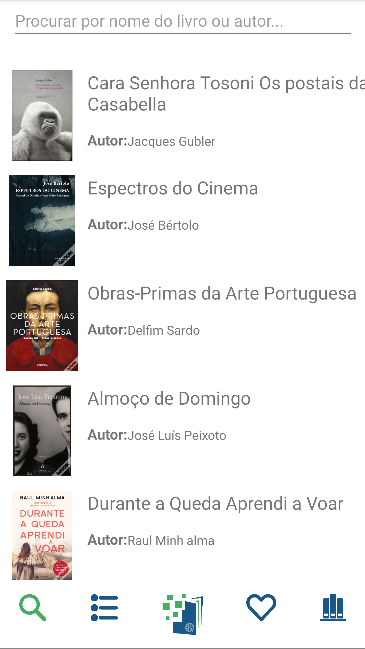


Figura - menu pricipal

# Interface da aplicação

**Menu procurar:**

* Nesta página o utilizador pode procurar um nome de um livro através de pesquisa por letras, facilitando assim a pesquisa de um livro cujo utilizador já sabe o nome do livro.
* À medida que o utilizador via escrevendo, aparece uma lista dos livros cujo têm essas letras no titulo do livro.

 Uma imagem com texto

Descrição gerada automaticamente

Figura -menu procurar 1/2

# Interface da aplicação

**Menu categorias:**

* Página com várias imagens com as diferentes categorias, o utilizador ao clicar na imagem é direcionado para uma nova página com uma lista de todos os livros dessa mesma categoria.
* Conta também com a barra de navegação, o ícone da página muda para verde para o utilizador saber em que página está.

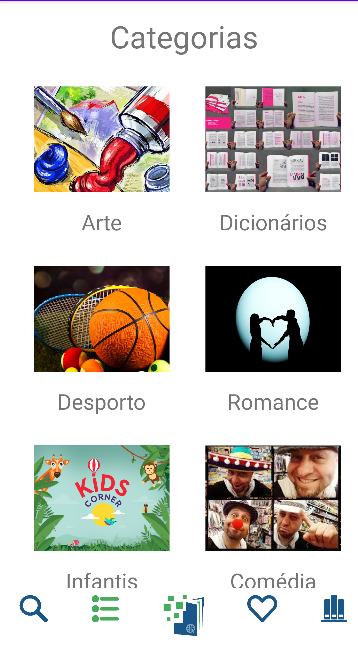


Figura -menu categorias

# Interface da aplicação

**Lista dos livros da categoria arte:**

* Lista com todos os livros da categoria anteriormente selecionada
* O utilizador ao clicar no livro desejado, vai ser direcionado para uma página com todos os detalhes sobre aquele livro:

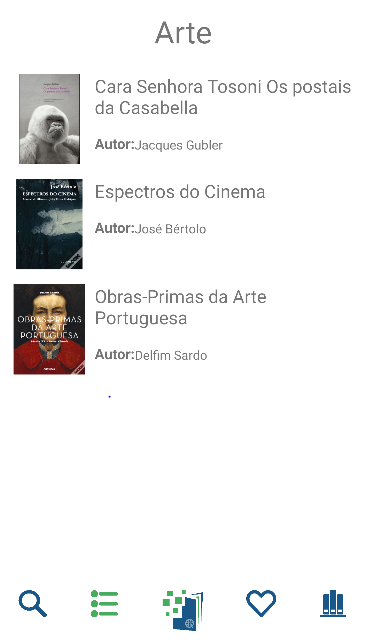


Figura - Lista arte

# Interface da aplicação

**Página com os detalhes do livro:**

* Página com todas as informações do livro, desde o título, o ano de lançamento, o nome do autor e a sinopse da obra.
* Em baixo contém dois botões que dá a possibilidade de o utilizador poder requisitar o livro ou então adicioná-lo aos favoritos.

Uma imagem com texto

Descrição gerada automaticamente

Figura -Livro detalhes

# Interface da aplicação

**Menu Favoritos:**

* Menu onde irá aparecer uma lista com todos os livros adicionados aos favoritos pelo utilizador.
* O utilizador ao clicar num livro desta lista, é direcionado para a activity anterior no relatório, que consiste na página com os detalhes do livro, onde na parte de baixo da página pode retirar o livro dos favoritos,

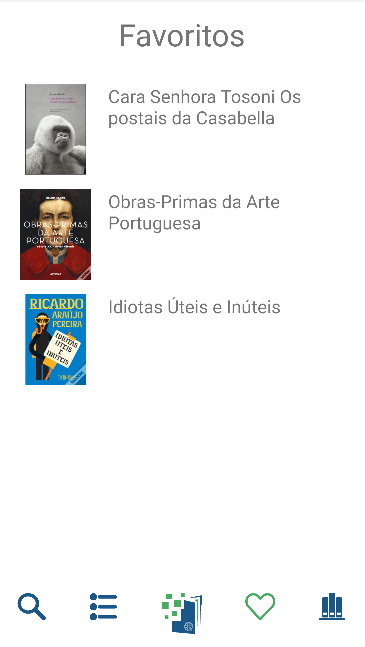


Figura -menu favoritos

# Interface da aplicação

**Menu Histórico:**

* Neste menu o utilizador pode rever a sua lista de livros requisitados ao longo do tempo.
* Esta página contém a lista dos livros, bem como o seu titulo e ainda tem a data de entrega.
* Ao ser selecionado vai abrir uma página com os detalhes do livro.

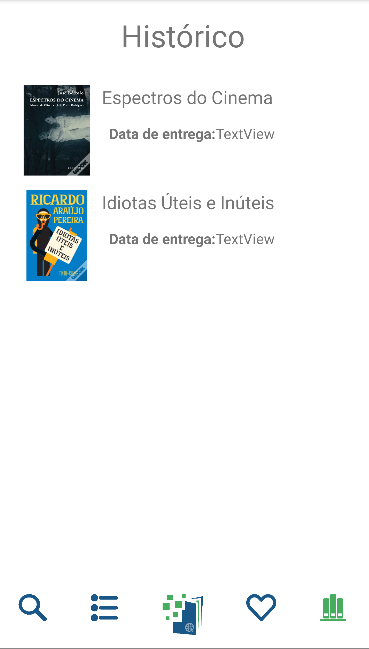


Figura - menu historico

# Interface da aplicação

**Menu Login:**

* Página onde o utilizador poderá efetuar o seu login para ter acesso ao resto da aplicação.
* Se não tiver conta, pode criar ao clicar no texto que diz para criar conta.

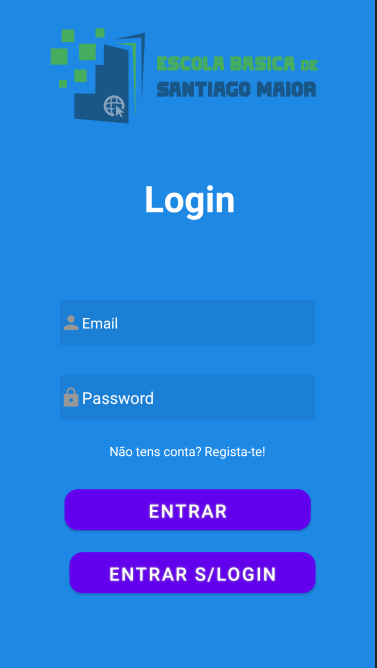


Figura -menu login

# Interface da aplicação

**Menu Registar:**

* Local onde o utilizador pode criar conta, só tem que preencher todos os requisitos pedidos e submeter.

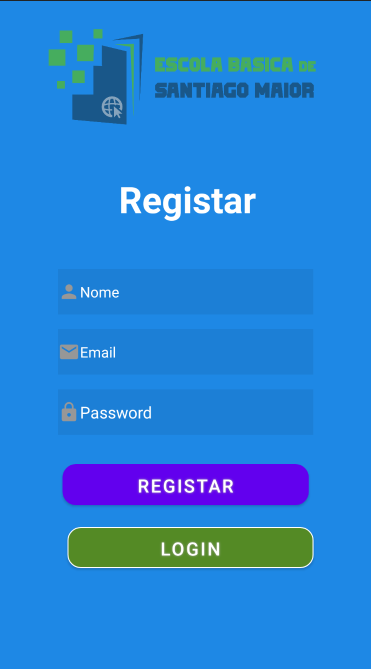


Figura -menu requisitar

# Testes com a aplicação

Com o programa finalizado, está na altura de fazer os testes. Os testes da aplicação, não são mais do que encontrar pequenos bugs que fazem com que a aplicação feche ou que o programa não consiga ler os caracteres que o utilizador digita.

No nosso caso houve algumas medidas que gostávamos de implementar na nossa aplicação, mas infelizmente não conseguimos, pois algumas delas não eram necessárias para o trabalho e outras ainda não tínhamos aprendido a forma de o fazer.

Fomos resolvendo pequenos bugs à medida que íamos criando a aplicação, pois também ao executá-la, a aplicação rebentava e tínhamos de concertar o erro.

O logcat do android studio deu imenso jeito, pois referia qual era a linha do código que fazia com que a aplicação rebentasse.

# Partes importantes do código

@Dao  
*public interface UserDao* {  
 @Query("SELECT \* FROM User")  
 *List*<User> getAll();  
  
 @Query("SELECT \* FROM User WHERE id = :id")  
 User getById(*long* id);  
  
 @Query("SELECT \* FROM User WHERE email = :email")  
 User getByEmail(String email);  
  
 @Query("SELECT \* FROM User WHERE email = :email and password = :password")  
 User checkLogin(String email, String password);  
  
 @Query("SELECT \* FROM User WHERE email = :email")  
 User checkRegister(String email);  
  
 @Insert  
 *void* add(User user);  
  
 @Insert  
 *void* add(*List*<User> users);  
  
 @Delete  
 *void* delete(User user);  
  
 @Update  
 *void* update(User user);  
}

* Local onde podemos criar as nossas querys em mysql.

# Partes importantes do código

*package* pt.ipbeja.estig.twdm.pdm1.project.models;  
  
*import* androidx.room.Entity;  
*import* androidx.room.PrimaryKey;  
*//falta o ano do livro  
//falta tambem o numero de requisiçoes*@Entity  
*public class* Book {  
 @PrimaryKey(autoGenerate = *true*)  
 *private long* id;  
 *private* String name;  
 *private* String description;  
 *private* String cover;  
 *private* String category;  
 *private* String author;  
 *private* String year;  
 *private int* reqNumber;  
 *private boolean* isFavourite;  
 *private boolean* wasReq;  
 *//private long date;  
  
  
 public* Book(*long* id, String name, String description, String cover, String category, String author, String year, *int* reqNumber, *boolean* isFavourite, *boolean* wasReq) {  
 *this*.id = id;  
 *this*.name = name;  
 *this*.description = description;  
 *this*.cover = cover;  
 *this*.category = category;  
 *this*.author = author;  
 *this*.year = year;  
 *this*.reqNumber = reqNumber;  
 *this*.isFavourite = isFavourite;  
 *this*.wasReq = wasReq;  
  
 }  
  
 *public boolean* isWasReq() {  
 *return* wasReq;  
 }  
  
 *public void* setWasReq(*boolean* wasReq) {  
 *this*.wasReq = wasReq;  
 }  
  
 *public boolean* isIsFavourite() {  
 *return* isFavourite;  
 }  
  
 *public void* setIsFavourite(*boolean* isFavourite) {  
 *this*.isFavourite = isFavourite;  
 }  
  
 *public void* setId(*long* id) {  
 *this*.id = id;  
 }  
  
 *public long* getId() {  
 *return* id;  
 }  
  
 *public* String getName() {  
 *return* name;  
 }  
  
 *public* String getDesc() {  
 *return* description;  
 }  
  
 *public* String getCover() {  
 *return* cover;  
 }  
  
 *public* String getDescription() {  
 *return* description;  
 }  
  
 *public* String getCategory() {  
 *return* category;  
 }  
  
 *public* String getAuthor() {  
 *return* author;  
 }  
  
 *public* String getYear() {  
 *return* year;  
 }  
  
 *public int* getReqNumber() {  
 *return* reqNumber;  
 }  
}

Pagina onde escrevemos todas as variáveis que constituem o livro.

# Partes importantes do código

*public class* BookAdapter *extends* RecyclerView.Adapter<BookAdapter.BookViewHolder>{  
 *private static final* String *TAG* = "BookAdapter";  
  
 *private List*<Book> bookList;  
 *private* Context context;  
  
 *public* BookAdapter(Context context, *List*<Book> bookList){  
 *this*.bookList = bookList;  
 *this*.context = context;  
 }  
  
 @NonNull  
 @Override  
 *public* BookViewHolder onCreateViewHolder(@NonNull ViewGroup parent, *int* viewType) {  
 Log.*i*(*TAG*, "onCreateViewHolder");  
 View view = LayoutInflater.*from*(*this*.context).inflate(R.layout.*list\_item*, parent, *false*);  
 *return new* BookViewHolder(view);  
 }  
  
 @Override  
 *public void* onBindViewHolder(@NonNull BookViewHolder holder, *int* position) {  
 Book book = *this*.bookList.get(position);  
 *//holder.getTextViewName().setText(book.getName());* Glide.*with*(*this*.context).load(book.getCover()).into(holder.getImageViewCover());  
 *//holder.getTextViewDesc().setText(book.getAuthor());* holder.getParentLayout().setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View v) {  
 Log.*i*("RecyclerViewAdapter", String.*format*("Clicked on: %s (position %d)", book.getName(), position));  
  
 DetailsActivity.*startActivity*(BookAdapter.*this*.context, book.getId());  
 }  
 });  
 }  
  
 @Override  
 *public int* getItemCount() {  
 *return this*.bookList.size();  
 }  
  
  
 *public class* BookViewHolder *extends* RecyclerView.ViewHolder{  
 *private final* View parentLayout;  
 *private* ImageView imageViewCover;  
 *private* TextView textViewName;  
 *private* TextView textViewDesc;  
  
 *public* BookViewHolder(@NonNull View itemView) {  
 *super*(itemView);  
 *this*.imageViewCover = itemView.findViewById(R.id.*imageView*);  
 *//this.textViewName = itemView.findViewById(R.id.bookName);  
 // this.textViewDesc = itemView.findViewById(R.id.bookDesc);  
 this*.parentLayout = itemView.findViewById(R.id.*parentLayout*);  
 }  
  
 *public* ImageView getImageViewCover() {  
 *return* imageViewCover;  
 }  
  
 *public* TextView getTextViewName() {  
 *return* textViewName;  
 }  
  
 *public* TextView getTextViewDesc() {  
 *return* textViewDesc;  
 }  
  
 *public* View getParentLayout() {  
 *return* parentLayout;  
 }  
  
 }  
}

Página do adapter, serve para apresentar os items na recyclerview defenida.

# Conclusão

Neste projeto acabamos por fazer uma aplicação que oferece aos utilizadores uma forma de requisitar livros disponíveis.

Cumprimos todos os requisitos que nos foram propostos, e tivemos um excelente trabalho em dupla.

Este projeto foi de grande importância para nós, pois fez-nos juntar tudo o que aprendemos durante o semestre nas disciplinas, Programação de dispositivos móveis e Sistemas interativos.

Concluímos assim o nosso projeto com excelentes bases nas linguagens Java com android Studio e com grande vontade para continuar a aprender mais.



Figura 14- Stickman a certificar tarefas

# Referências Bibliográficas

<https://developer.android.com/studio?gclid=Cj0KCQjw5auGBhDEARIsAFyNm9FLk8GasacIcEmpeQDMiR1jAXPgssrli4Oe-r8ypbfX6GTFMgNLRhoaApBeEALw_wcB&gclsrc=aw.ds>

https://central.github.com/deployments/desktop/desktop/latest/win32

<https://adobe-xd.softonic.com.br/download>

<https://www.adobe.com/pt/products/illustrator/free-trial-download.html#mini-plans-web-cta-illustrator-card>

Exercícios lecionados nas aulas de sistemas interativos

Exercícios lecionados nas aulas de programação para dispositivos móveis

<https://developer.android.com>

<https://pt.stackoverflow.com>