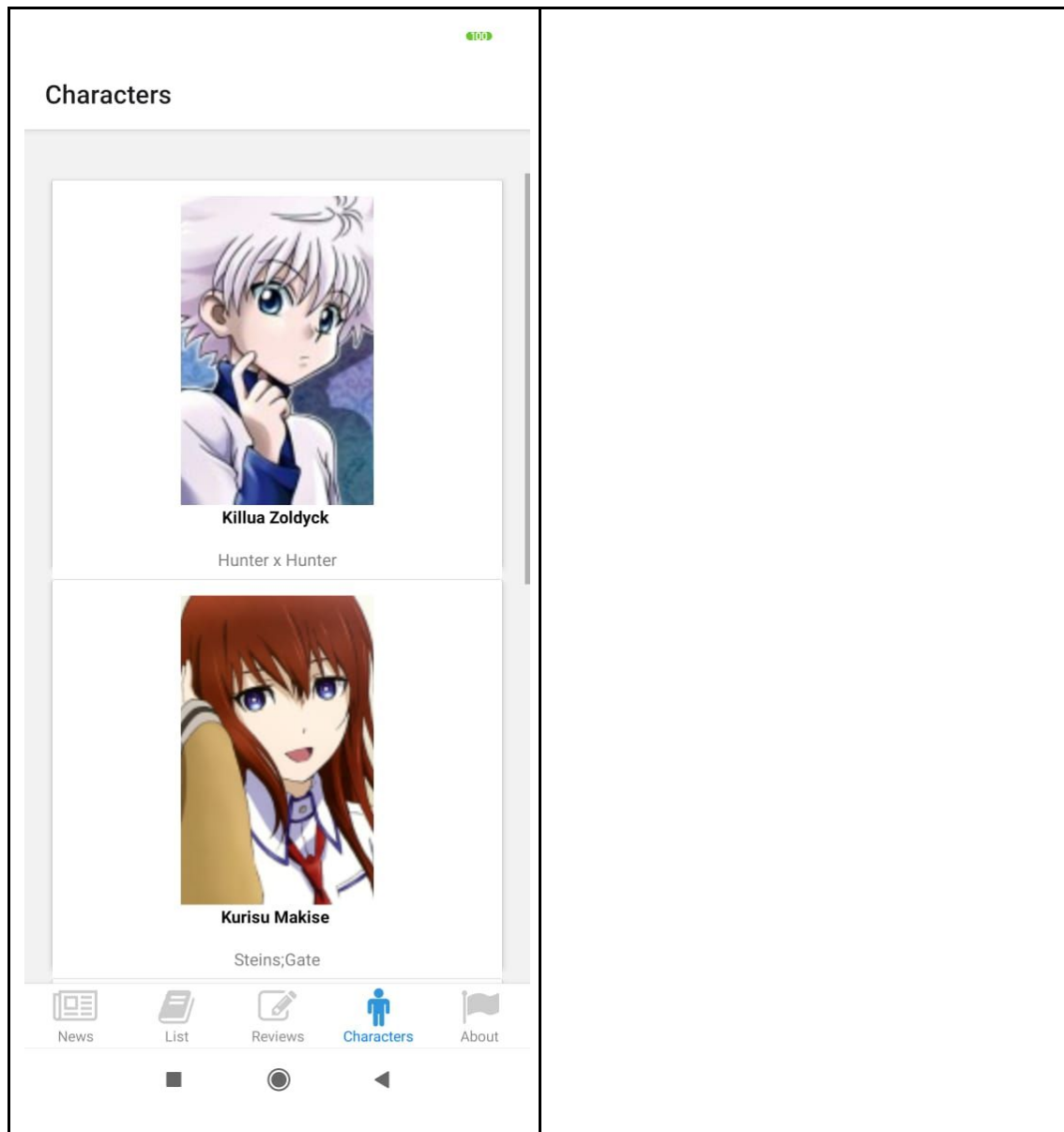


 <p>CEFSa Centro Educacional da Fundação Salvador Arena</p>	<p>Código: ECP 744</p> <p>Disciplina: Linguagem de Programação 3</p> <p>N2 1º bimestre Curso: Eng. De Computação Turma: EC6</p> <p>10/04/2020</p> <p>Prof.: Daniel Carvalho Coord.: Michele Basana</p>	
Aluno: Lucas Boulle	RA: 081170012	Nota:
Aluno: David Conde	RA: 081170002	
Aluno: Filipe Marques	RA: 081170007	
Aluno: Rodrigo Candido	RA: 081170031	
Aluno:	RA:	
<p>Orientações:</p> <ul style="list-style-type: none">• Este trabalho deve ser feito em grupos de 4 alunos. Não podendo ter de forma alguma, número maior ou menor de alunos; (Haverá um grupo de 3 alunos pois estamos em número primo, 27)• Caso o grupo não seja composto da forma estabelecida será atribuída nota zero;• Caso tenham trabalhos iguais, ambos terão nota zero;• A nota do trabalho será a nota de N2 do 1º bimestre;• O trabalho deverá ser postado no moodle, na disciplina LPIII, apenas por um integrante do grupo e deverá conter o nome e RA de todos os componentes;• O trabalho a ser entregue deverá ser postado, IMPRETERÍVELMENTE, até, no máximo, dia 10/04, sexta-feira às 23h55min. Após este horário e data, o moodle estará travado para receber os trabalhos e a nota será zero;• A não realização do trabalho fará com que o aluno tenha nota zero na N2 do 1º bimestre;• Qualquer dúvida sobre problemas na postagem entre em contato com a coordenação do curso, por isso, teste antes do dia 10/04;• Abaixo critérios de correção: <p>Descrição do trabalho: Elaborar um projeto de aplicativo novo utilizando tecnologia front-end WEB: HTML5, CSS3, JavaScript, Bootstrap, Cordova e consumir WEB API disponível na Internet (back-end). O aplicativo deve implementar 4 "features", páginas HTML cada uma delas deve consumir uma WEB API diferente. Trabalhem em Pair Programming remoto via Skype, Hangouts, Zoom, WebEx e a fins... Este projeto explora o nível cognitivo CRIAR conforme a taxonomia de Bloom. O código fonte será avaliado e revisado no GitHub.</p> <p>Rubrica: Nota máxima para os grupos que conseguirem gerar o .APK e rodar o aplicativo no Android (ou iOS) usando Apache Cordova, vale 1 ponto. Cada uma das 4 telas (páginas HTML) do aplicativo será avaliada com nota entre 0 a 2,25, esta nota será composta entre a média de: design, complexidade, qualidade e coragem*</p> <p>Links:</p> <p>Aplicação <i>template</i> (modelo – usar como base do projeto): https://github.com/danielscarvalho/FTT-Cordova-LP3-Template</p> <p>Aplicação de referência: https://github.com/danielscarvalho/FTT-Cordova-LP3-N1</p> <p>Lista de WEB APIs: https://github.com/public-apis/public-apis</p> <p>* Extreme Programming Explained, Kent Beck, Cynthia Andres, 2005, Addison-Wesley, 2ª Ed, Cap 4</p>		

Link do GitHub com a aplicação desenvolvida:

Feature 1 breafing: Descrever a *feature* implementada, fornecer o link da API...

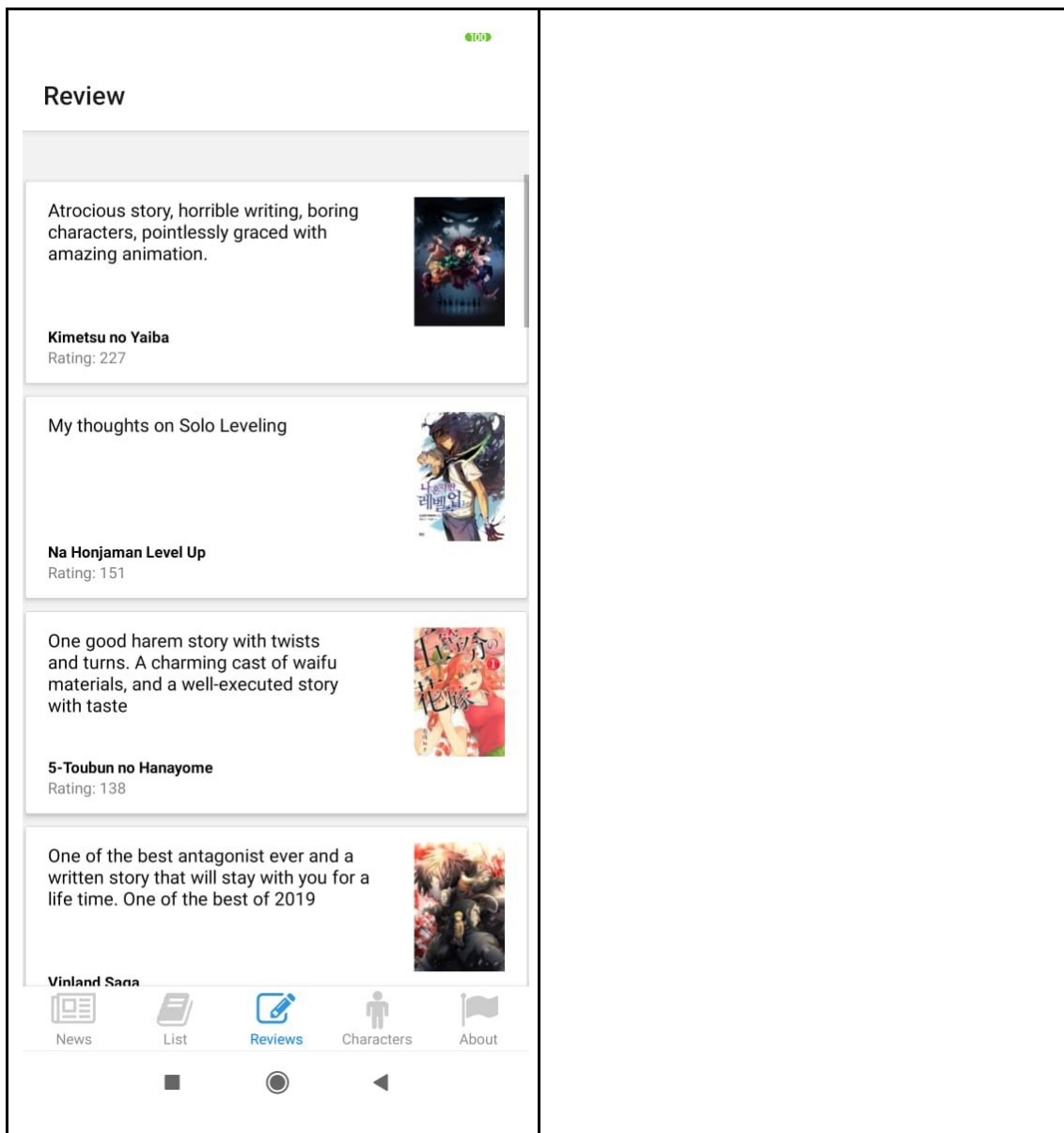
Feature que busca dados de personagens na api anilist registrados na base de review do site



Feature 2 breafing: Descrever a *feature* implementada, fornecer o link da API...

Fornecer duas telas (*printscreen*) do aplicativo como evidência:

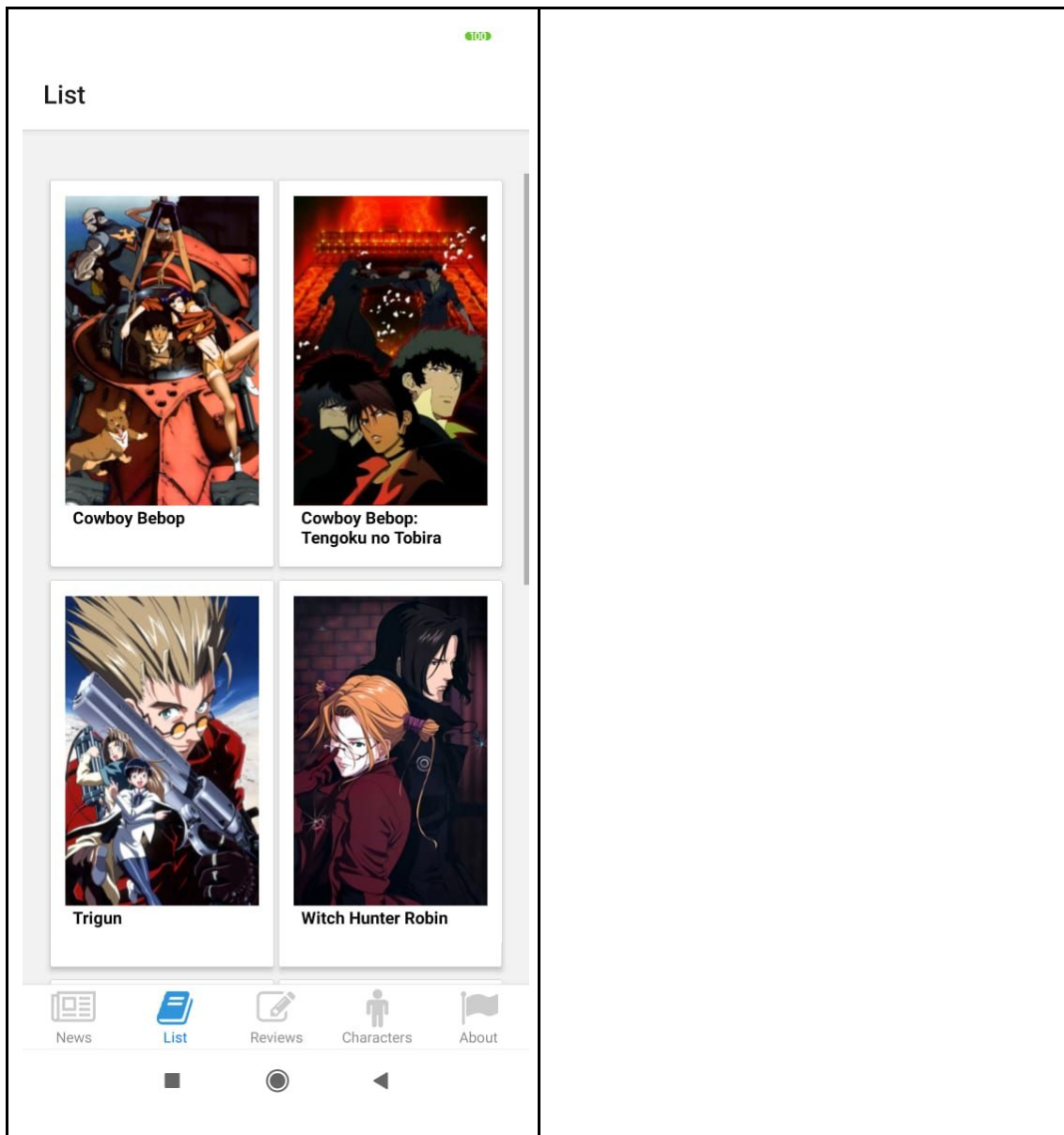
Feature que busca rating com review da comunidade anilist disponibilizada pela própria responsável



Feature 3 breafing: Descrever a *feature* implementada, fornecer o link da API...

Fornecer duas telas (*printscreens*) do aplicativo como evidência:

Feature que traz todos os animes cadastrados na base da anilist



Feature 4 breafing: Descrever a *feature* implementada, fornecer o link da API...

Fornecer duas telas (*printscreen*) do aplicativo como evidência:

Feature que busca threads abertas a qualquer usuário na anilist

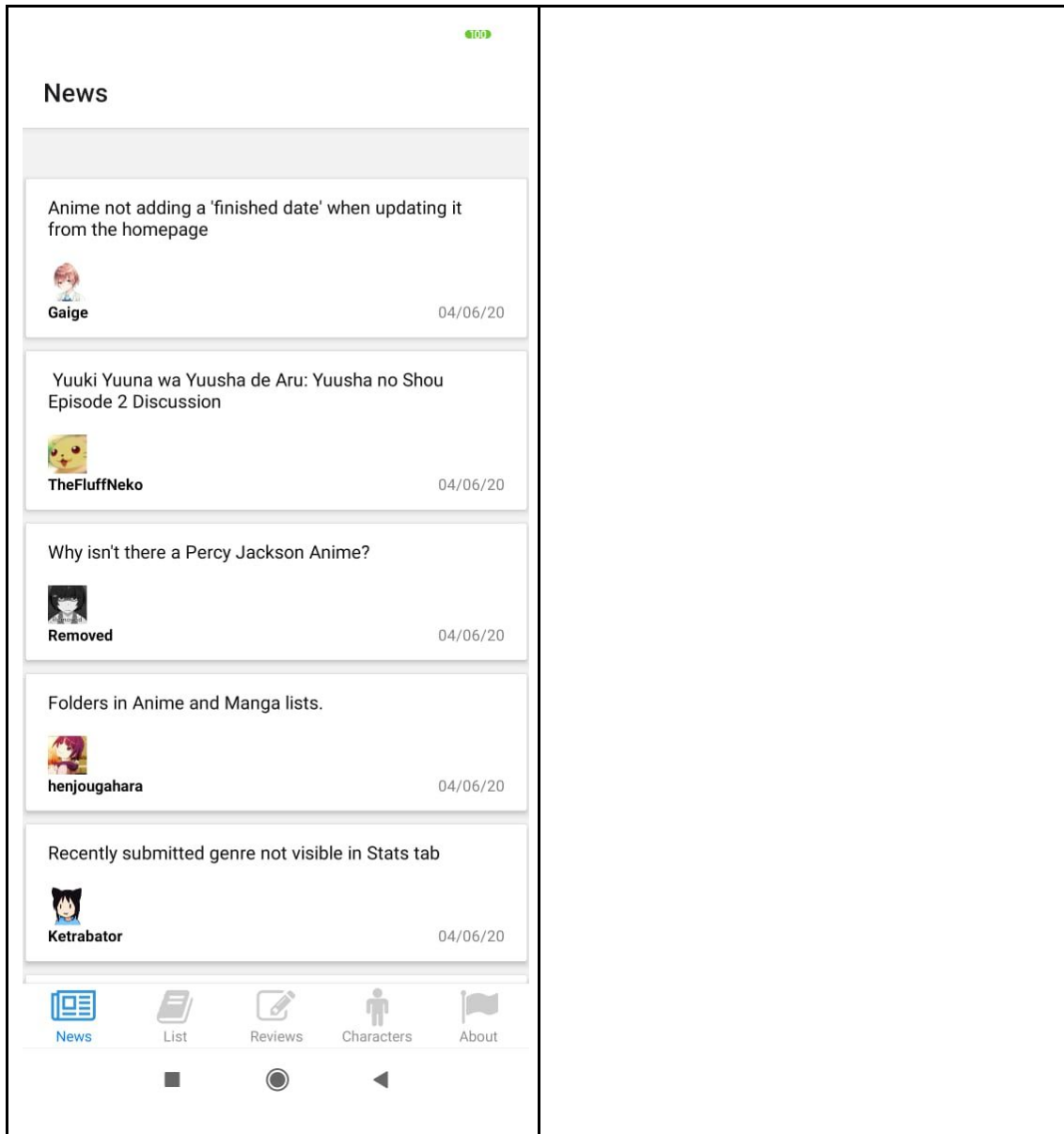
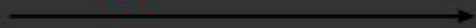


Diagrama: Elaborar um gráfico para ilustrar como é feita a chamada de WEB API:

 React Native



GraphQL



Service request

ANILIST