

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Phone version:](#)

[Login screen](#)

[Front page/All posts feed](#)

[Subreddit feed](#)

[Post screen \(comments\)](#)

[Subreddit search](#)

[Subscription settings](#)

[Settings screen](#)

[Clickable phone prototype](#)

[Tablet](#)

[Widget](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

**GitHub Username:** tisqui

# Just Read for Reddit

## Description

This application is a reader for the popular social news site Reddit for people who want just read, not contribute. Reddit is a overwhelming, huge content database, origin of a lot of internet memes. It has a lot of features and large API support(like a swiss knife). Users generate tons of the new content every minute.

### Main hypothesis

Most of the reddit reader, available now are focusing on implementing as many features as they can, providing dozens of settings. This is great for the socially active users, who want to comment, moderate, participate, etc. They are explorators. But what about those who are mostly browsing only selected subreddits for years and hardly posted 10 comments and upped 50 posts? They need some lightweight app which won't make them frustrated and fulfill their main need - read the reddit they are used to.

**Sub-hypothesis:** Reddit is probably equally valuable by it's post's content and the comments to the posts. In some subreddits the comments are even more interesting than the post. So the app should be focused on the comfortable comments reading, not the fast posts access (like swiping between pages to switch the posts). Users do not want to read the random post, they click to open only posts they like.

### Solution

Application allow users to construct the set of subreddits to read and then forget about these settings - just read the posts sorted by the settings they applied. Reading and browsing experience should be seamless. Long comments threads should be readable. Users should not have a lot of settings on the appearance (just the day/night mode), caching, updating - because the users of this app want to relax. Decide what is better for them without asking them. The images/videos content should be available for 1 click from the list of posts.

Application will have a login, so user can share the subscriptions configuration with the website and subscriptions will be saved. Also there will be a search for the subreddits in case user want to explore new ones.

## Intended User

This is the application for the existing Reddit users, who are:

- more conservative than explorative
- overwhelmed by the amount of subreddits and the new content
- want a simple way to consume their favorite subreddits; make the settings and filters configurations once and enjoy the content every day.
- like my husband.

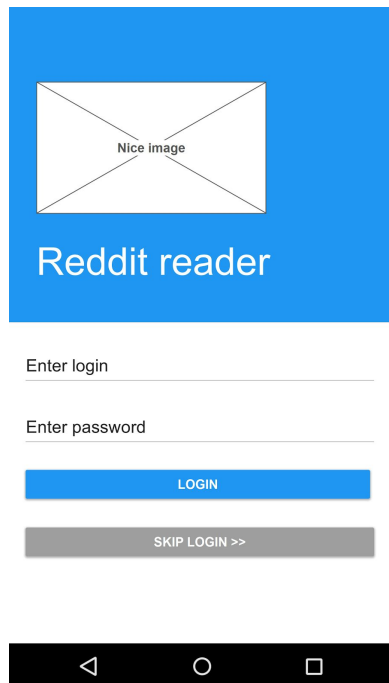
## Features

1. Login
2. Frontpage posts feed, "All" posts feed
3. Subreddit posts feed
4. Subreddit information panel
5. Drawer menu
6. Settings
7. Day/night UI themes
8. Subreddits subscription configuration
9. Search for subreddit
10. Subscribe/unsubscribe to subreddit
11. Filter subreddits feed
12. Pagination for the posts feed
13. Read post comments
14. Share post on Twitter and Facebook
15. See the post media content (image/video) in webview
16. Vote for post - Up/down

## User Interface Mocks

### Phone version:

#### Login screen



#### Starting screen.

Login screen is the starting screen of the application.

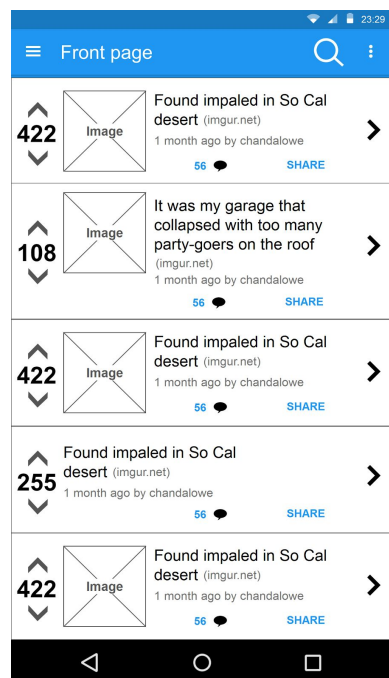
User can skip the login pressing “Skip login” - then application will apply the default “Reddit picks” subscription configuration.

User should use their Reddit login/password. If the login/password are not correct - show popup with error.

Logged out user will be asked to login and forwarded to this screen in the following situations:

- Subreddit subscription
- Voting for post

#### Front page/All posts feed



Page with the feed of the “Front page”/”All” feeds of the Reddit. Does not have the right “Informational” subreddit panel.

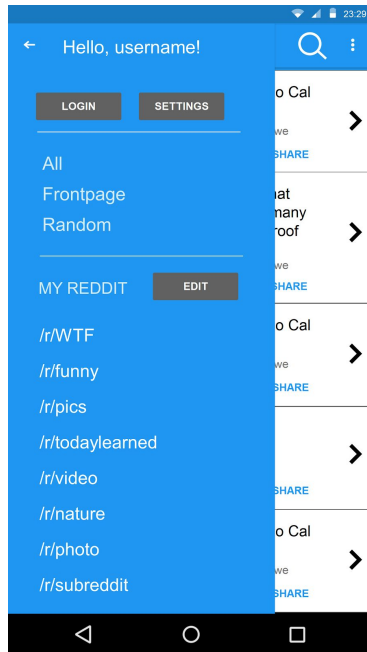
User can click:

- Hamburger menu - opens the drawer menu
- Search icon - opens the search screen
- Menu - opens the filtering menu

Feed has pagination - 20 posts per each page. When user scrolls to the bottom - next posts are uploaded.

In the feed element user can click:

- Voting up/down
- Title of the post - open media content/comments
- Comments - open comments
- Share
- Arrow - open comments



### Drawer menu:

Opens when user pressed the hamburger menu or swipe right.

Closes when user swipes left or presses icon.

On the top - "Hello, [username]" where the [username] is replaced by the login. If the user is logged off - it is stated as "username".

### Clicks:

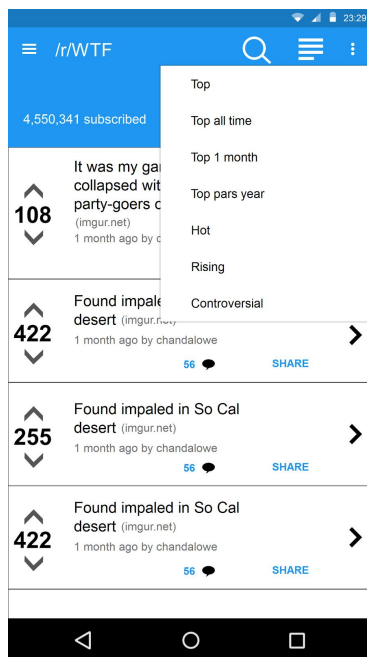
Login button - opens login screen

Settings button - opens settings screen

All/Frontpage/Random - opens Frontpage feed screen

Items from "My reddit" - opens subreddit feed screen

Edit button - opens subscriptions edit screen

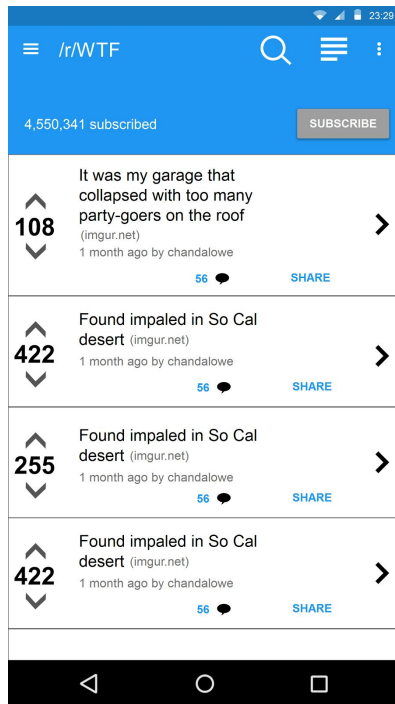


### Filter menu:

- Hot
- New
- Rising
- Controversial
- Top
- Top: past 24 hours
- Top: past week
- Top: past month
- Top: past year
- Top: all time

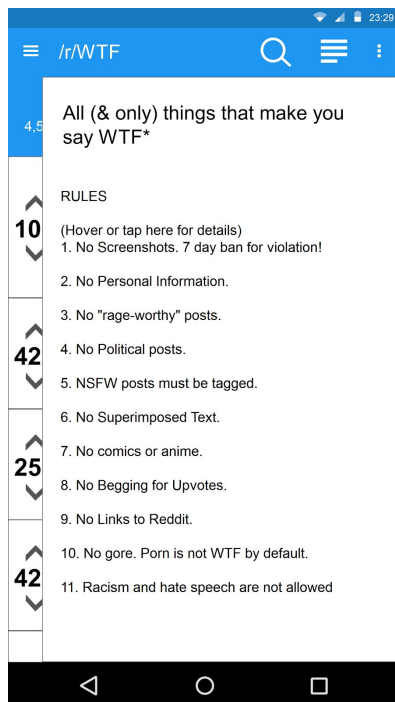
When user selects the new filter - the list reloads.

## Subreddit feed



The same as the Frontpage feed, but:

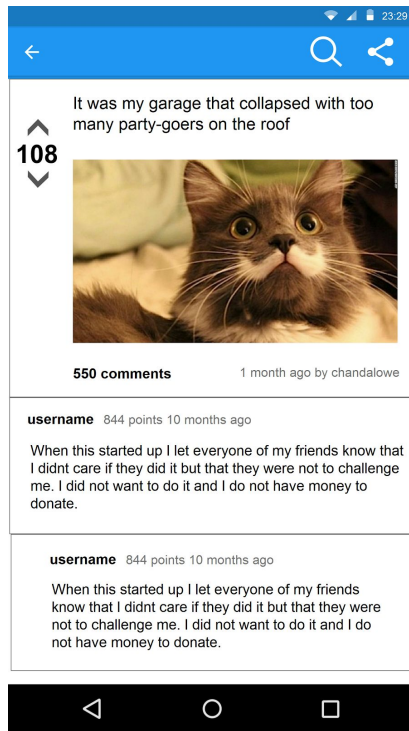
- Has the taller toolbar, which contains number of subscribers and “Subscribe” button. When user scrolls - the toolbar collapses, additional info disappears.
- Has the “subreddit info” panel, opened by the icon on the top right



### Subreddit info panel:

Contains the rules of the subreddit and any information the subreddit moderators/owners post there.

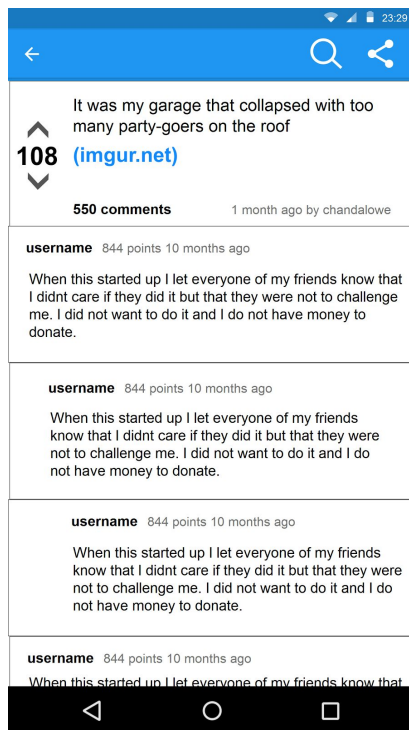
## Post screen (comments)



Post with embedded media content:

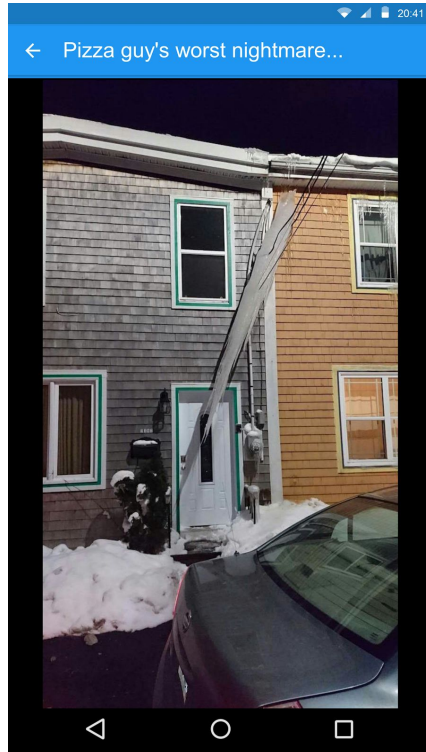
- Title of the post
- Media content (image/video preview)
- Number of votes
- Vote up/down buttons
- Number of comments
- Time of the post
- Comments feed

Comments will be visualized showing the dependency(depth).



Post without embedded media content:

- Has the name of the content origin website, which is clickable and follows to the webview with the media content.



Web view with the media content.

## Subreddit search



Start searching when user hits "enter" on the keyboard. Not a live search.

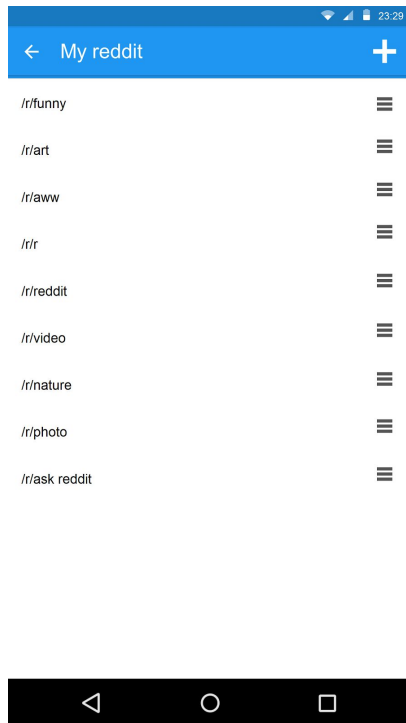
Search results:

- List of subreddits
- Button "subscribe" for each
- On click - opens subreddit





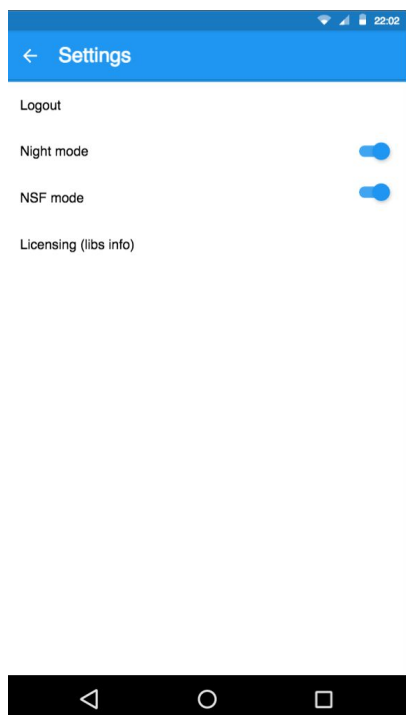
## Subscription settings



User can:

- Drag and drop the subreddits to sort them
- When tapping the subreddit item in the list - the menu opens with options to subscribe the subreddit
- On tapping “+” - search of the subreddits opens

## Settings screen



Settings screen includes:

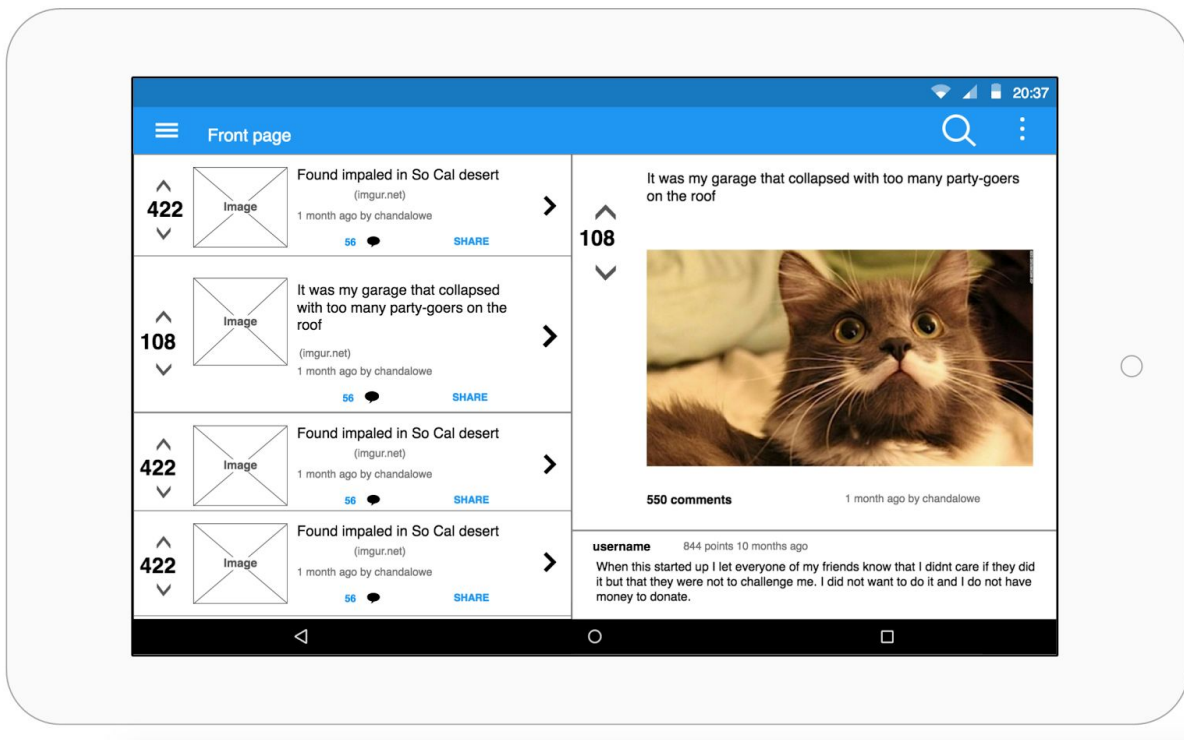
- Logout: ask the user if he really wants to logout - dialog. If yes - logout the user.
- Switch on/off the night mode. Application will contain 2 color schemes - for the daytime - light and the dark to the night mode.
- NSF mode - switched on the Not Safe for Work filter (im demo version will be on by default)
- Licensing info - text page with the information about the libs (will need to study the terms/licencing of the libs I am using)

## Clickable phone prototype

Here is the clickable prototype, created in Invision:

<https://squisqui.proto.io/share/?id=cd4533ac-6239-401a-abd8-d0abf2d86174&v=3>

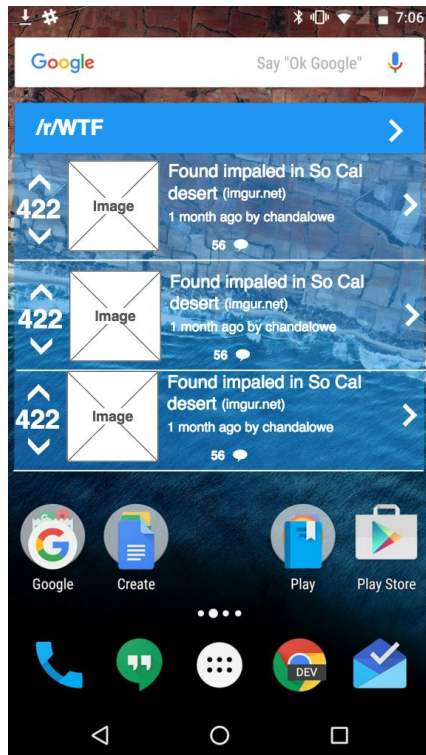
## Tablet



The only screen which will have the significant difference from the phone version will be the feed screen - it is split in 2 panels. The left panel contains the feed of posts. The right - the post details and the comments.

All the other screens will be adjusted to the tablet - fonts sizes, elements sizes, margins, etc. But the layout composition will remain the same.

## Widget



Widget should have 2 possible sizes - medium - 1x3, large size is 1x4 cells (and expandable by height).

Widget will show the latest selected subreddit posts

Widget will have control to choose the subreddit to show

On click on the list item - open the post

Update of the widget should be scheduled.

## Key Considerations

### How will your app handle data persistence?

#### **Store with ContentProvider:**

App will use the ContentProvider to store the latests posts and subscriptions in database (SQLite). Latests posts will be available for display them even when offline.

#### **Sync with SyncAdapter:**

App will regularly pull and send data to/from API, update data in its cache at regular intervals using a SyncAdapter

#### **Loaders:**

App will use a Loader to move its data to its views.

### Describe any corner cases in the UX.

#### **Front page screen**

- When user opens the right panel - close the drawer and vise a versa
- When no data in the list - show the empty view
- When server error - show the user friendly message to the user
- Do not reload list when screen rotated
- When user gets back to this screen from the post detailed view - leave the scroll on the same position, not the beginning of the list

#### **Subreddit feed screen**

- The same as above
- When user scrolls - collapse the navigation bar to the standard size, hide subscription button and number of subscribers

#### **Subscription settings screen:**

When user presses "Back" ask the user in a dialog if they want to save the settings.

#### **Drawer**

When the user hits back - close the drawer

#### **Login**

- Do not force the user to log in
- Check for login for - subscribe/unsubscribe, getting subscriptions, upvote the post. Show the dialog.

Describe any libraries you'll be using and share your reasoning for including them.

Reddit API: <https://www.reddit.com/dev/api>

Title	Link	Reasoning
JRAW	<a href="https://github.com/thatJavaNerd/JRAW">https://github.com/thatJavaNerd/JRAW</a>	API wrapper To simplify the work with Reddit API
Butterknife	<a href="http://jakewharton.github.io/butterknife/">http://jakewharton.github.io/butterknife/</a>	Field and method binding
Glide	<a href="https://github.com/bumptech/glide">https://github.com/bumptech/glide</a>	Media management and image loading
SugarORM	<a href="http://satyan.github.io/sugar/">http://satyan.github.io/sugar/</a>	To simplify the interaction with SQLite database

## Next Steps: Required Tasks

Decomposition of the feature list in technical tasks.

**Development approach:** implement the complete features, one by one, so they can be tested, not “UI first, functionality after”. First features will also contain some basic fundamental “logic” parts, as data layer, sync adapter, etc. The last features on the list will use the existing infrastructure and take less time.

**Design:** Will use one of the provided material color schemes, material guidelines and material pack icons. No custom design.

N	Task	Subtasks
1	Project setup	<ul style="list-style-type: none"> <li>• Create project</li> <li>• Configure libraries</li> <li>• Clean the basic project selected</li> <li>• Create project module structure</li> <li>• Set permissions</li> </ul>
2	Frontpage feed	<ul style="list-style-type: none"> <li>• Create basic data layer for posts</li> <li>• Create api request processing to get data for post</li> <li>• Create basic styles for the app</li> <li>• Create toolbar layout</li> <li>• Make the preparations for the tablet layout</li> <li>• Create RecyclerView for feed list</li> <li>• Create layout for list item</li> <li>• Create adapter for the list</li> <li>• Create loader</li> <li>• Connect UI to the data layer - controllers</li> <li>• Implement initial version of SyncAdapter</li> <li>• Implement “Load more”</li> <li>• Save the downloaded posts locally</li> <li>• Add the “Empty list” view</li> <li>• Handle “No internet connection”</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Handle screen rotation</li> <li>• Create tests</li> </ul>
3	Frontpage feed tablet support	<ul style="list-style-type: none"> <li>• Create the UI</li> <li>• Make the necessary changes in functionality</li> <li>• Handle screen rotation</li> </ul>

4	Drawer menu	<ul style="list-style-type: none"> <li>• Create UI</li> <li>• Add the navigation to the existing pages</li> <li>• Handle screen rotation</li> <li>• Tablet support</li> </ul>
5	Open the post media content	<ul style="list-style-type: none"> <li>• Get the link of the content from the post list request</li> <li>• Create the web view which will show the media content</li> <li>• Add the navigation</li> <li>• Add the “No internet connection” case</li> <li>• Add adjustments to the tablet version UI</li> <li>• Handle screen rotation</li> </ul>
6	Detail post page	<ul style="list-style-type: none"> <li>• Create the UI for the page</li> <li>• Create 2 UI cases - with the media content embedded and the media content in webview on the new page</li> <li>• Handle screens transition</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Add adjustments to the tablet version UI</li> <li>• Handle screen rotation</li> <li>• Create tests</li> </ul>
7	Comments to the post	<ul style="list-style-type: none"> <li>• Implement the API request</li> <li>• Implement saving data locally</li> <li>• Add the UI for the comment post element</li> <li>• Create the recyclerview</li> <li>• Make posts to have the different position based on the “level”</li> <li>• Add “load more” to load more posts when user scrolls</li> <li>• Handle screen rotation</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Add adjustments to the tablet version UI</li> <li>• Create tests</li> </ul>
8	Subreddit feed	<ul style="list-style-type: none"> <li>• Implement request to get the subreddit feed page 1</li> <li>• Store the a data locally</li> <li>• Implement the UI base on the Front page layout</li> </ul>

		<ul style="list-style-type: none"> <li>• Implement load more for the list</li> <li>• Handle screen rotation</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Add adjustments to the tablet version UI</li> <li>• Create tests</li> </ul>
9	Right subreddit info panel	<ul style="list-style-type: none"> <li>• Add the API request handling</li> <li>• Create UI</li> <li>• Adjust UI to the tablet</li> <li>• Handle screen rotation</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Create tests</li> </ul>
10	Login feature	<ul style="list-style-type: none"> <li>• Implement the API request</li> <li>• Implement UI</li> <li>• Validate user input</li> <li>• Handle screen rotation</li> <li>• Add the skipping of login</li> <li>• Link the navigation to the login to every existing relevant place</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Add adjustments to the tablet version UI</li> <li>• Create tests</li> </ul>
11	Subreddit search	<ul style="list-style-type: none"> <li>• Add the subreddit search API request</li> <li>• Create the UI for the search panel</li> <li>• Create the UI for the search results</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Add adjustments to the tablet version UI</li> <li>• Create tests</li> </ul>
12	Subscribe to the subreddit	<ul style="list-style-type: none"> <li>• Implement the API request to subscribe/unsubscribe</li> <li>• Check if user is logged in</li> <li>• Changed the locally saved data</li> <li>• Add the subscribe action to the search results</li> </ul>



		<ul style="list-style-type: none"> <li>• Add the subscribe to the subreddit feed page</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Create tests</li> </ul>
13	Subscriptions settings	<ul style="list-style-type: none"> <li>• Add the API request processing for getting the subscriptions</li> <li>• Add the UI for the screen</li> <li>• Store the subscriptions data locally</li> <li>• Add the ability to the list to drag and drop sorting elements</li> <li>• Store the sorting order locally</li> <li>• Add the menu for the each list element, which opens on tap</li> <li>• Add the search icon on the navigation which opens search page</li> <li>• Add the “No internet connection” case when trying to update the subscription. Use the stored version, if no - show message that no information available.</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Add adjustments to the tablet version UI</li> <li>• Create tests</li> </ul>
14	Settings	<ul style="list-style-type: none"> <li>• Create the Settings UI</li> <li>• Save the settings to the shared preferences</li> <li>• Check the settings when performing relevant functions</li> <li>• Add the licenses info screen</li> <li>• Add the logout function</li> <li>• Implement the request for NSW filter. Sort the content when the filter is on.</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Handle screen rotation</li> <li>• Handle tablet UI</li> <li>• Create tests</li> </ul>
15	Create 2 color schemes - day/night	<ul style="list-style-type: none"> <li>• Add the color schemes</li> <li>• Add the setting on the Settings screen</li> <li>• Implement for all UI when setting is changed</li> </ul>

		<ul style="list-style-type: none"> <li>• Create tests</li> </ul>
16	Filters menu	<ul style="list-style-type: none"> <li>• Create the UI</li> <li>• Create the API requests (add setting the filters to the feed requests)</li> <li>• Store the selected filter locally</li> <li>• Apply the filters to the feed</li> <li>• Handle screen rotation</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user</li> <li>• Create tests</li> </ul>
17	Share the post	<ul style="list-style-type: none"> <li>• Add sharing to Facebook, check for logged in. Sharing data: title of the post, source, link to the content.</li> <li>• Add sharing to Twitter, check for logged in. Sharing data: title of the post link to the content.</li> <li>• Handle offline mode</li> <li>• Handle server errors</li> </ul>
18	Upvote/downvote the post	<ul style="list-style-type: none"> <li>• Create the API request processing</li> <li>• Add the “No internet connection” case</li> <li>• Handle server errors - communicate to the user, add Try again button</li> <li>• Create tests</li> </ul>
19	Google analytics	<ul style="list-style-type: none"> <li>• Integrate with Google services</li> <li>• Implement the analytics functionality</li> </ul>
20	Google Ads	<ul style="list-style-type: none"> <li>• Implement showing the test adds on the main Frontpage feed screen in the bottom</li> </ul>
21	Create widget	<ul style="list-style-type: none"> <li>• Create widget UI</li> <li>• Create widget logic</li> <li>• Create widget update logic</li> <li>• Implement widget settings in Settings section (what subreddit to show)</li> </ul>
22	Local cache	Limit the stored data, but make sure user has smth to see if we previously opened the app
23	Accessibility	<ul style="list-style-type: none"> <li>• Content descriptions</li> <li>• Focus based navigation</li> </ul>

		<ul style="list-style-type: none"> <li>• Navigation using a D-pad</li> </ul>
24	Perform beta-testing with husband	-
25	Implement user feedback	-
26	Create a signed apk	-
27	Drink champagne!	-