How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
2. Name your document file: “**Capstone_Stage1**”
3. Replace the text in green

GitHub Username: tomasmichael995

Greek Podcasts

Description

Greek Podcasts app will attempt to fill the gap for the the greek podcasting community all around the world. Now there is a place to search and listen to content that is spoken solely in the hellenic language.

And it does get better. Greek Podcasts allows users to create their own shows and share their concerns with the world.

Note to the reviewer: The app will not use any external API to fetch podcasts. Sadly I could not find one with greek spoken content. So the idea is to be the one that provides it.

Intended User

This app targets users who want to listen to/create podcasts spoken in the greek language.

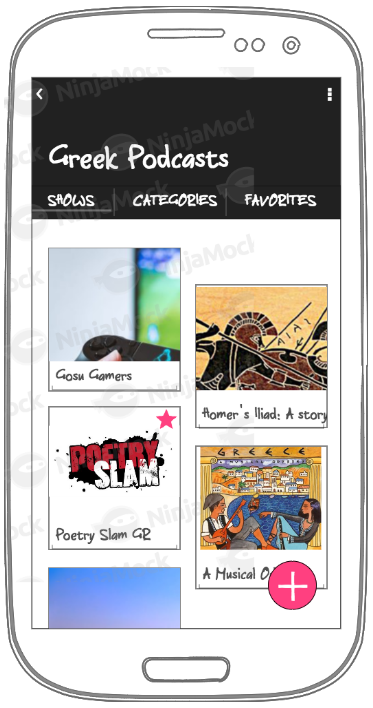
Features

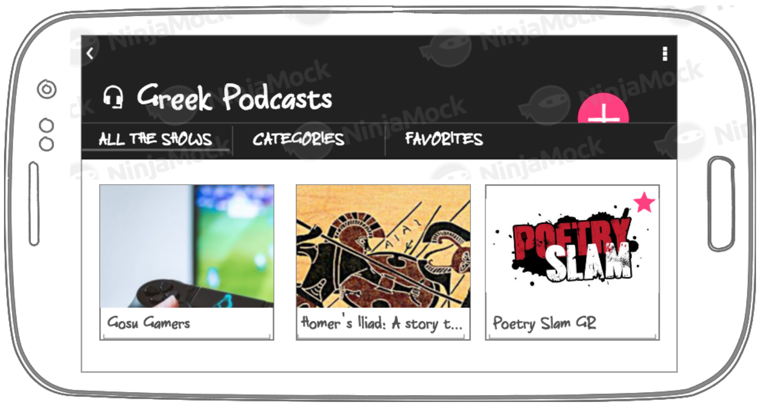
- Search podcasts
- Play podcasts
- Notifications when new content is uploaded
- Content downloading for offline listening
- Podcast creation
- Includes adds

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

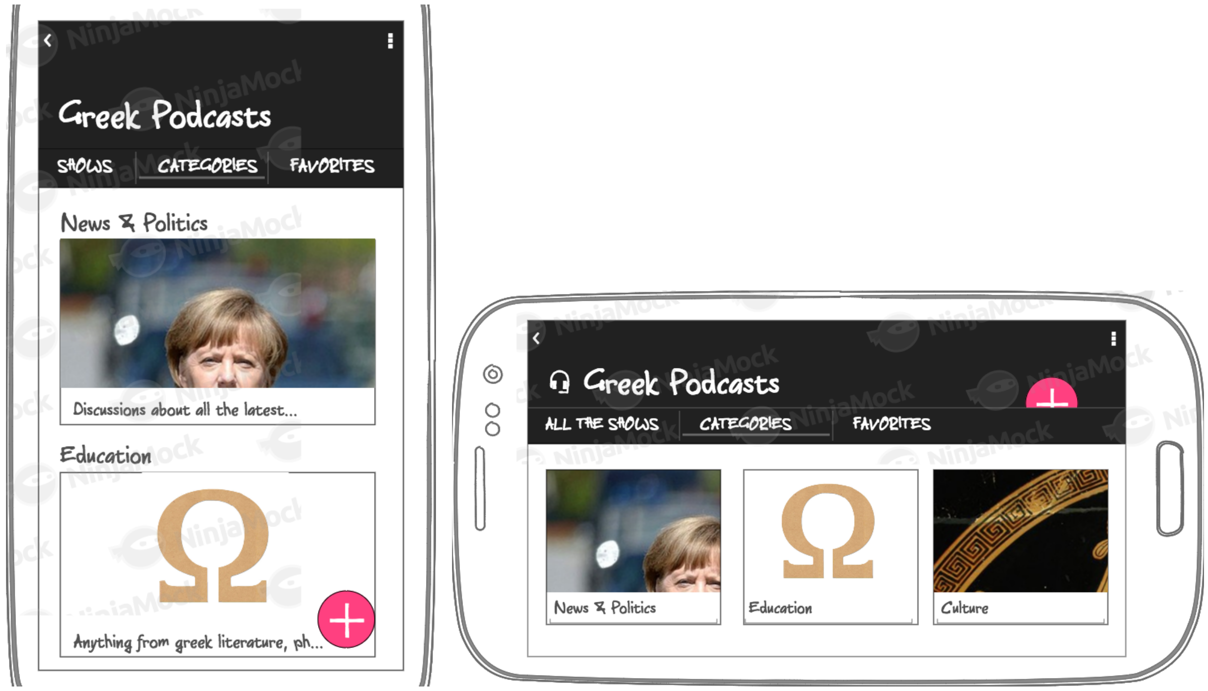
Screen 1





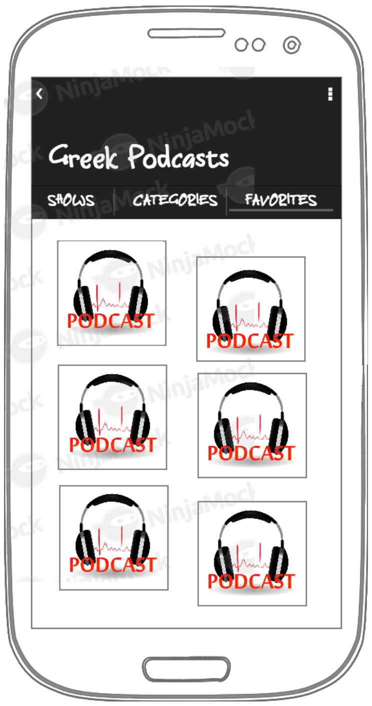
This screen is shown to the user when he opens the app. All the podcasts of the app are being displayed. The main action is the creation of a podcast (FAB). The pink star that is anchored on the top right corner of a podcast indicates that the latter is being followed by the user.

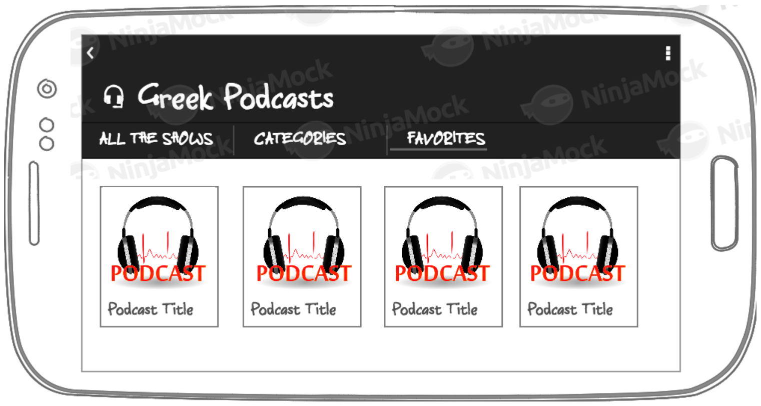
Screen 2



The middle tab of the main screen. It displays the categories where podcasts belong, in a nice way.

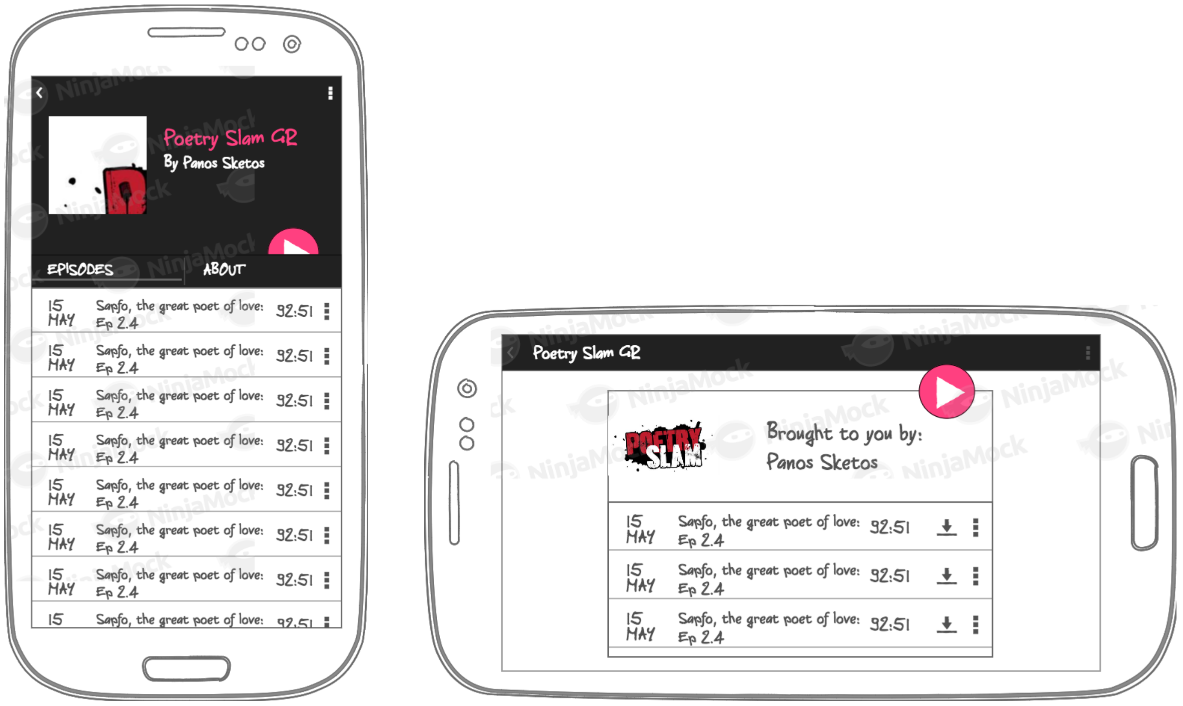
Screen 3





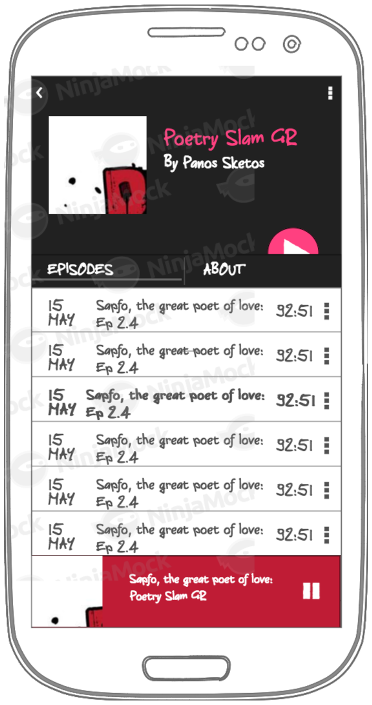
The third and last tab of the main screen. All the starred podcasts from the user are displayed here.

Screen 4



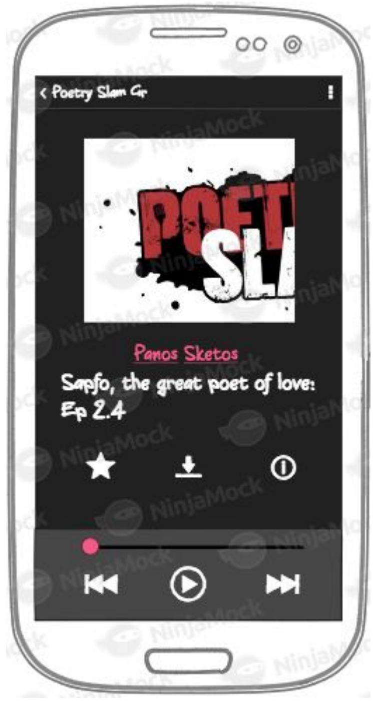
This screen previews a podcast show. It's episodes in the current tab and some info about the show in the second tab.
The main action is the 'play episode' action which can be achieved by the FAB (I apologise if sometimes I failed to bring the FAB to the front).
Besides playing an episode a user is able to download it locally to watch later offline.

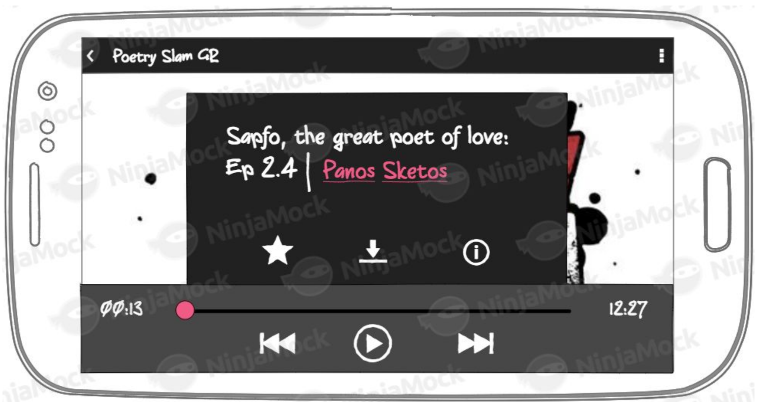
Screen 5


Poetry Slam GR
By Panos Sketos
EPISODES
ABOUT
15 MAY Sapfo, the great poet of love: Ep 2.4 92:51
15 MAY Sapfo, the great poet of love: Ep 2.4 92:51
15 MAY Sapfo, the great poet of love: Ep 2.4 92:51
15 MAY Sapfo, the great poet of love: Ep 2.4 92:51
15 MAY Sapfo, the great poet of love: Ep 2.4 92:51
15 MAY Sapfo, the great poet of love: Ep 2.4 92:51
Sapfo, the great poet of love: Poetry Slam GR

The same screen as previously, showing the playing bar at the bottom.
The episode item in the list that is currently being played is marked in bold typeface.
The color of the 'play bar' is red because it is the most common color of the show's image (picked by a color palette).

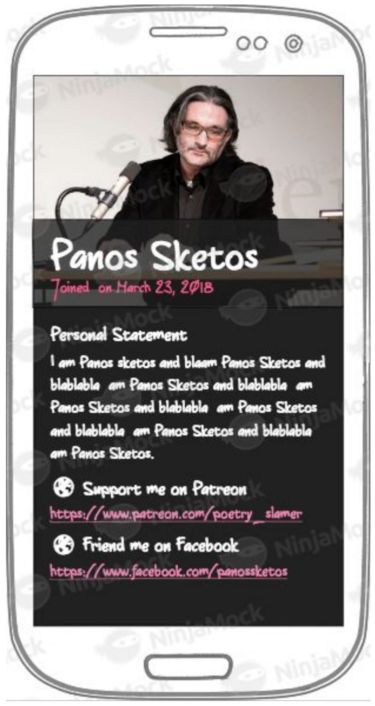
Screen 6





A separate screen just for playing an episode of a podcast.
Here the user can star (follow, which means he will be notified with new content) the show, download the episode or view some information about it (probably in a dialog). In this screen he has the ability to move forwards/backwards the episode or move to the next/previous one. Lastly the creator's name is displayed by a purple link. By clicking it he will be navigated to the next screen.

Screen 7



Panos Sketos

Joined on March 23, 2018

Personal Statement

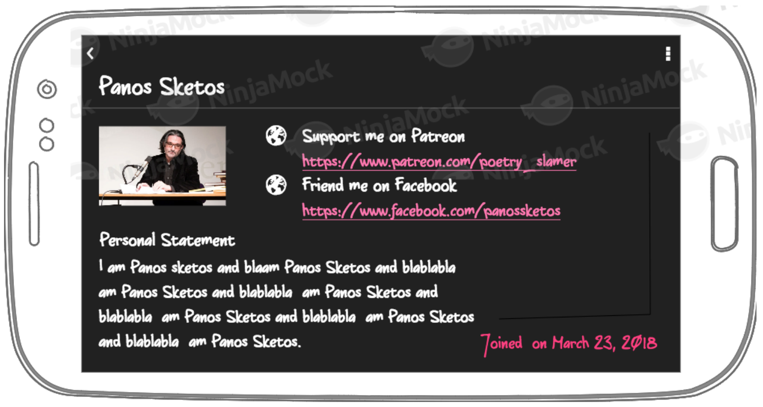
I am Panos sketos and blaam Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos.

Support me on Patreon

https://www.patreon.com/poetry_slamer

Friend me on Facebook

https://www.facebook.com/panossketos



Panos Sketos

Support me on Patreon

https://www.patreon.com/poetry_slamer

Friend me on Facebook

https://www.facebook.com/panossketos

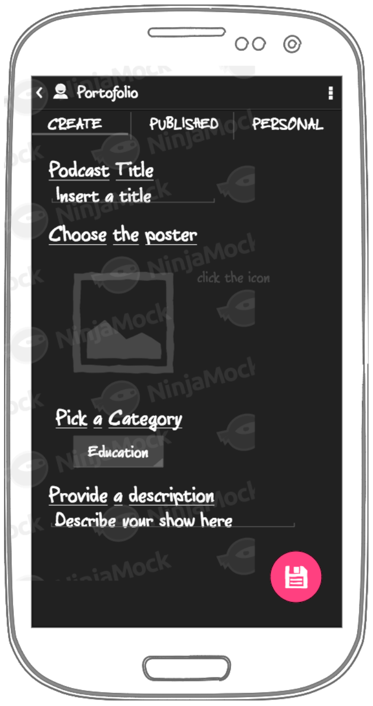
Personal Statement

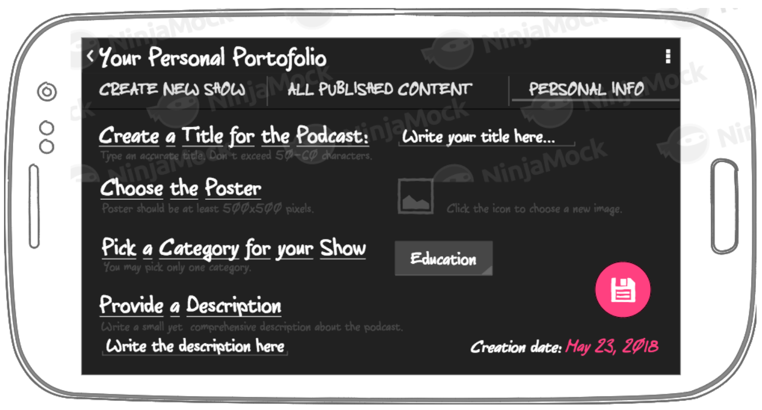
I am Panos sketos and blaam Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos and blablabla am Panos Sketos.

Joined on March 23, 2018

A podcaster's screen. The main purpose is to give information about the podcaster of course, but also gives the ability to promote him, through the "promotion links".

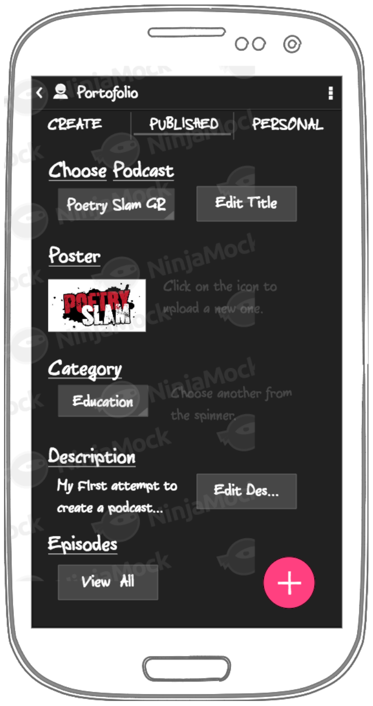
Screen 8

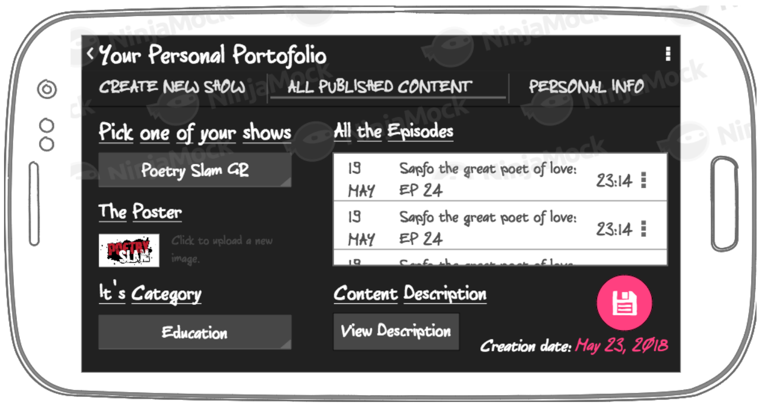



Your Personal Portofolio
CREATE NEW SHOW
ALL PUBLISHED CONTENT
PERSONAL INFO
Create a Title for the Podcast:
Write your title here...
Type an accurate title. Don't exceed 50-60 characters.
Choose the Poster
Poster should be at least 500x500 pixels.
Click the icon to choose a new image.
Pick a Category for your Show
You may pick only one category.
Education
Provide a Description
Write a small yet comprehensive description about the podcast.
Write the description here
Creation date: May 23, 2018

Podcaster's portofolio.
When a user wants to create a podcast then he will be navigated to the above screen (after a succesful login).
In the current tab the user/podcaster must provide a title, a poster and other essential content to create a new, yet empty, podcast.
The main action represented by the FAB here is the 'save content' action. This gives the user the power to take a moment, ensure that the content is accurate and save when he feels ready to.

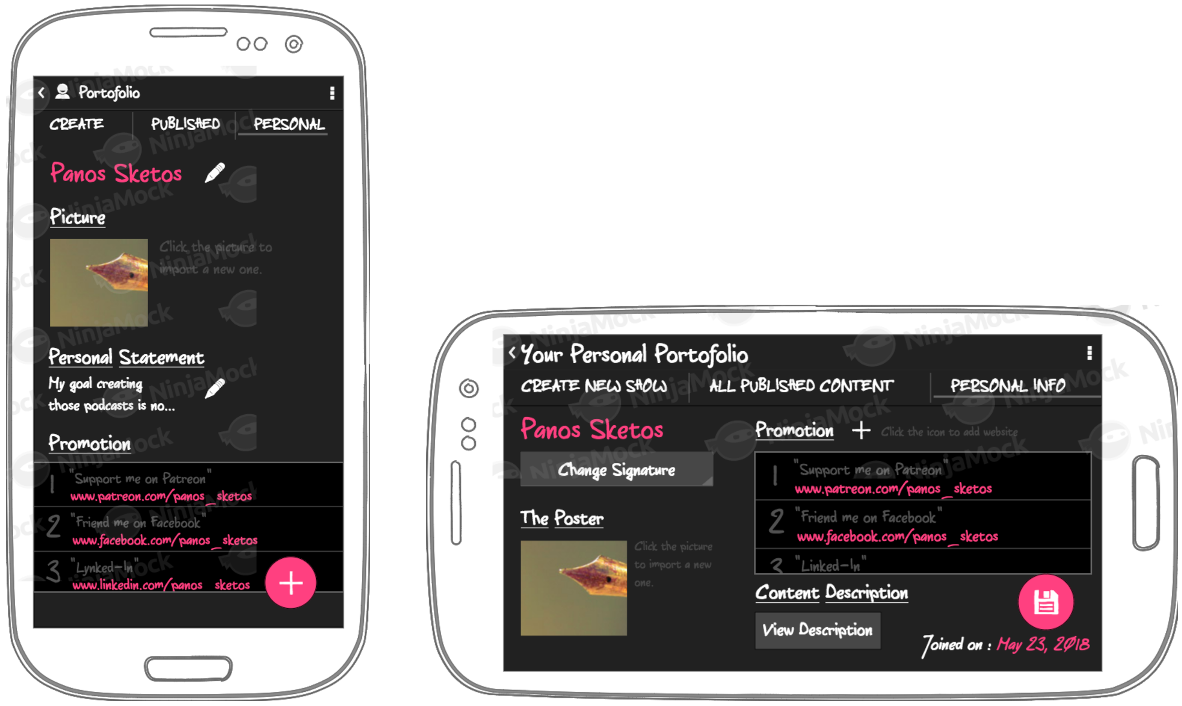
Screen 9





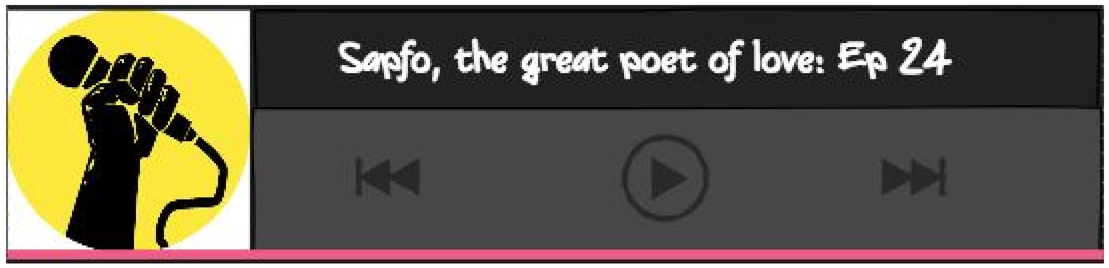
Same screen as the one previously.
The current tab is the middle one and displays all the published content of the podcaster.The podcaster may choose his available show from the spinner, he may upload a poster for it, change its category etc.
The main action is the same, the 'saving action' as the FAB indicates.

Screen 10


Portofolio
CREATE
PUBLISHED
PERSONAL
Panos Sketos
Picture
Click the picture to import a new one.
Personal Statement
My goal creating those podcasts is no...
Promotion
1 "Support me on Patreon" www.patreon.com/panos_sketos
2 "Friend me on Facebook" www.facebook.com/panos_sketos
3 "Lynked-In" www.linkedin.com/panos_sketos
Your Personal Portofolio
CREATE NEW SHOW
ALL PUBLISHED CONTENT
PERSONAL INFO
Panos Sketos
Change Signature
The Poster
Click the picture to import a new one.
Promotion
Click the icon to add website
1 "Support me on Patreon" www.patreon.com/panos_sketos
2 "Friend me on Facebook" www.facebook.com/panos_sketos
3 "Linked-In"
Content Description
View Description
Joined on : May 23, 2018

Third and last tab of the portofolio screen. The podcaster edits all his personal info that is being displayed to the user. Again the main action stays the same, save the content at will.

Screen 10



The app widget displays the podcast's poster and the episode that was lastly being played. The user may continue the play or click the poster to navigate again to the app.

Key Considerations

How will your app handle data persistence?

I will build a Content Provider that will persist data locally. This data will be user info such as starred podcasts, episodes he watched etc.

On the contrary podcasts and podcasters data will be stored online using Firebase Database (for metadata) and Firebase Storage for the necessary audio/image files etc.

Lastly, I probably need to use SharedPreferences for app utilities.

Describe any edge or corner cases in the UX.

For example, how does the user return to a Now Playing screen in a media player if they hit the back button?

- Screen 3: When a user tries to remove a starred podcast.
- Screen 8/9/10: If the user exits without hitting the main action, which is the saving operation then he should be notified or be asked to confirm the save.
- When the user is asked to upload an image of himself or a poster for the podcast. It shouldn't be too large or too small in size.

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

- ButterKnife for binding views.
- Picasso to handle the loading and caching of images.
- The android-support-design library for implementing material design in the app.
- The android-support-palette library to draw a view taking an image as a reference.
- Exoplayer as the main media player for the app.

Describe how you will implement Google Play Services or other external services.

I will implement Google Sign-In. I will save users email at Firebase Database to make a distinction between users.

I am also planning to use AdMob. I have never used it before so I am not able to describe properly how it will be implemented. One thing certain is that it will exist on the free flavor of the app.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Firstly I will implement and configure all the external libraries.

Task 2: Implement UI

I am going to implement the MVC architectural pattern. And as a result in this approach the activities are considered to be 'controllers' and they will not draw their layout directly, but for the through the 'MVC View'.

Potato, potatoe, I will implement UI for all the screens above.

Task 3: Database Schema

Taking under consideration the use cases of the app I have to design database schemas. Those are:

- A database schema for persisting users data locally.
- A database schema for persisting podcasts and podcasters data online (Firebase Database).
- Not a database schema, but a file structure. How the podcasts will be saved etc.

Task 4: Testing Firebase

I have worked with Firebase Database before but not with the Storage service. As I result I need to study the documentation and implement it to confirm that everything works as expected.

Firebase in a Weekend might also be a good idea.

Task 5: Create Mock Data

Instead of implementing Firebase services right away I will first create a class that provides fake data in order to test aspects of the UI that need this data.

This data could be for example podcasts that need to be shown in the main screen.

Task 6: Main Screen Implementation

The creation of the first screen. The (fake) podcasts must be fetched from network and be displayed on screen. Also categories as well as favorites.

Not sure if I should try to handle a case where there is no network connection, because Firebase services include offline capabilities.

Task 7: Podcast Screen Implementation

Creation of the screen that displays all the podcasts. Podcast (fake) info must be fetched and bind to the views.

Task 8: Play an Episode

This is the first attempt using the ExoPlayer. If I click an episode (item in the list) then the player's fragment must appear and play the episode of course.

- Refresh ExoPlayer documentation.
- Show the media player when an episode is clicked.
- Play the episode's audio file (first search the file locally and then online)

Task 9: Play Screen Implementation

Apart from view binding which will be a common task for all the screens, this screen needs to implement new functionalities such as star a podcast, download it to be available online etc.

So the subtasks here are:

- Star a podcast.
- Learn how to save files locally.
- Implement the saving functionality.
- Media player must play the episode from where the user left it.

Task 10: Broadcaster's Screen Implementation

Just binding views and open browser when a user clicks a link.

Task 11: Google Sign-In

Before moving to the creation of the new podcast, which is the main action of the first screen, the user must identify himself by signing in with his google account.

I guess this functionality cannot be mocked so it's time to implement it.

Task 12: Portofolio Screen Implementation

Apart from the view binding there are task such as:

- Update all content when user hits the save FAB (for now it will be fake operation, later content would be updated into Firebase servers).
- The save operation (the FAB actually that performs it) must be deactivated until an image file is being downloading.
- Set a proper animations for the FAB button, considering the above scenarios.
- On personal tab, implement add website functionality.

Task 13: Search Podcasts

Every screen should be providing a direct way to search for podcasts by name. While the name is being typed by the user a fragment could pop-up to display the queried podcasts while clicking one will navigate the user to the Podcast Screen (Screen 4 in the mock-ups).

Task 14: From Fake to Real Data

Now it's time to switch from the fake data providers to the real ones such Firebase.

Task 15: Notifications

This step could be done earlier, before step 14. I don't know which way is better. Anyway, a JobScheduler may run once or twice per day and check if new episodes are available for the starred podcasts. If there are new, then a notification will be popped to the user informing him about the fact.

Clicking the notification navigates directly to the Podcast Screen (Screen 4 in the mock-ups).

Task 16: Advertisement

I have never implement AdMob so I will have to study documentation. The adds must be implemented in the free variant of the app.

Submission Instructions

- After you've completed all the sections, download this document as a PDF [File → Download as PDF]
    - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"