Destiny Casts Project Outline

GitHub Username: noahpatterson

DestinyCasts

Description

DestinyCasts surfaces weekly podcasts about the popular video game Destiny. Including a set list of the most popular Destiny podcasts, DestinyCasts is your first stop to get the weeks latest news and commentary about the game. No longer will you favorite Destiny podcasts be buried among your other podcasts. With DestinyCasts is a clutter free, distraction free way to get what you want. More Destiny.

Intended User

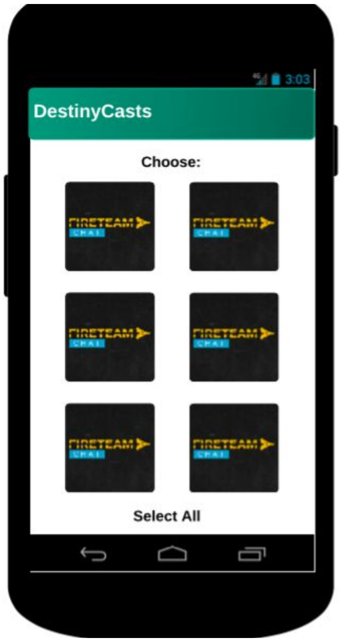
Fans of the video game Destiny.

Features

- Curated list of top destiny podcasts
- Simple lists of “this week's” and “last week’s” episodes
- Ability to choose which podcasts to follow
- Podcast and episode descriptions
- Widget to access the latest episode of your favorite shows

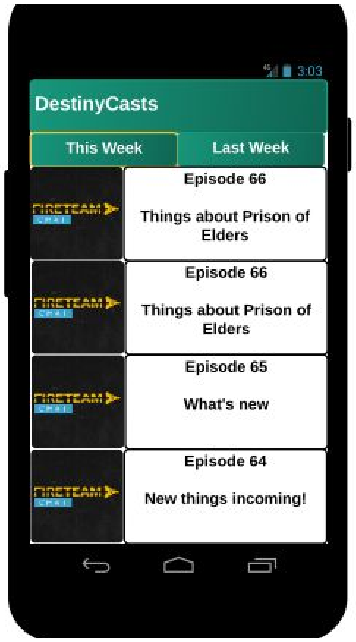
User Interface Mocks

Choose/Intro Screen:



Screen to choose which podcasts you want in your list. List is saved. You can swap podcasts later from this screen.

Episodes By Week



Episode 66

Things about Prison of Elders

Episode 66

Things about Prison of Elders

Episode 65

What's new

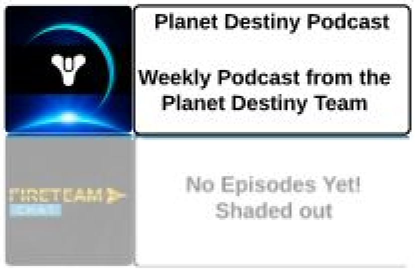
Episode 64

New things incoming!

Tabs contain “this week’s” and “last week’s” episodes respectively. Episodes sorted by most recent. Contains all current episodes from each of your selected favorites.

Detail about episodes in weekly list

Detail List is sorted by shows that have episodes this week.



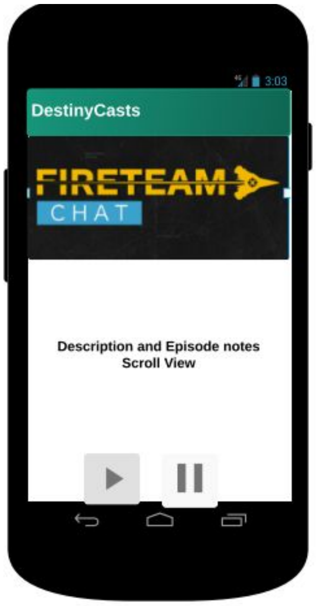
Planet Destiny Podcast

Weekly Podcast from the Planet Destiny Team

No Episodes Yet!
Shaded out

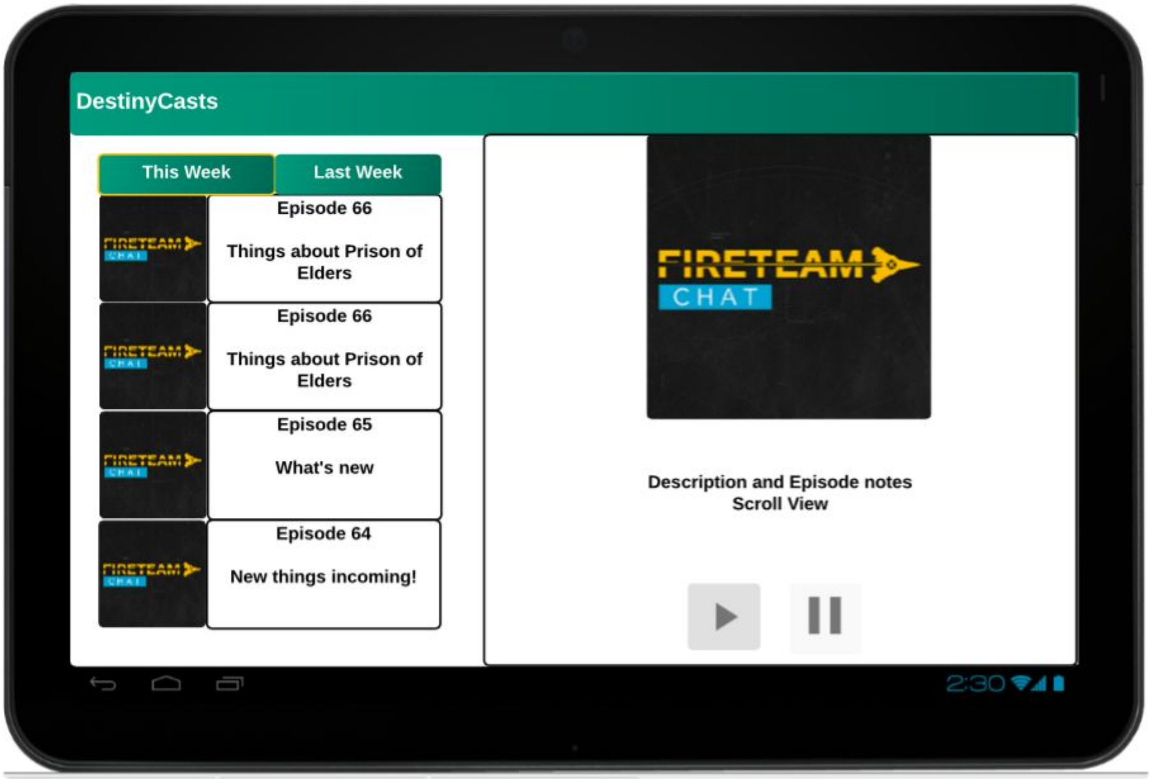
If the show doesn't yet have an episode we gray out the image and text.

Episode View and Play controls



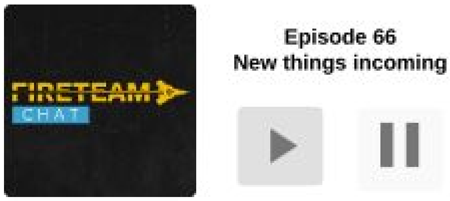
Episode view that provides episode information and ability to play the episode.

Tablet master/detail view



Widget

Widget with Most recent episode



Key Considerations

How will your app handle data persistence?

I will build a Content Provider that pulls data from RSS feeds and stores it. Data is persisted till the app restarts (and can connect to the internet) or the user manually refreshes. I will only pull 10 episode for each podcasts at a time.

Describe any corner cases in the UX.

I will need to either provide a now playing button and highlight the playing podcast in order for the user to navigate to the playing podcast.

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

Picasso to handle the loading and caching of images.

Next Steps: Required Tasks

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

Task 1: Project Setup

- Configure App
- Gather list of RSS feeds
- Setup Glyde, admob and google analytics

Task 2: Implement UI for Each Activity and Fragment

- Build UI for IntroChooseActivity
- Build UI for WeeklyListActivity
- Build UI for PodcastPlayerActivity
- Build UI for PodcastDetailActivity
- Build tablet views

Task 3: Build IntentService for RSS feed data

- Gather a list of RSS feeds
- Build xml parser to grab needed fields

Task 4: Build Content Provider to store RSS data

- Create table structure including
    - Podcast date: date podcast was released
    - **Next week date:** date 7 days from podcast date that podcast episode moves to "last week"
    - **Expire date:** date 14 days from podcast date that the episode expires
    - **Week:** podcast episodes current week, 0 if no weeks, 1 this week, 2 last week
- Sync IntentService data pull to content storage
- Create query and retrieval calls

Task 5: Hookup Intro/Choose screen

- Store choices in saved preferences
- Link to WeeklyListActivity

Task 6: Populate WeeklyList from ContentProvider

- Place determined podcasts into the right week

- Link each podcast to EpisodeDetailView
- Work out weekly list sorting by has episode this week. Epidsode was updated in the since Monday of this week.

Task 7: Add admob interstitial

- Place between WeeklyList and EpisodeDetailView every 4th time.

Task 8: Hookup EpisodeDetailView and Player

- Simple player only has play/pause
- Only streams music for now

Task 9: Build Widget

- Add current week's track with play pause button when not playing.
- Should also update to currently playing track when playing.

Task 10: Error handling, Network states and edge cases

- Double check all error handling, network states and edge cases

Task 11: Add Google analytics

- To all key metric touch points

Task 12: Prep for release

- Sign keys, build apk

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"