



netidee

PROJEKTE

RichPods.org

Zwischenbericht | Call 19 | Projekt ID 7391

Lizenz: CC BY-SA

Inhalt

1	Einleitung.....	3
2	Status der Arbeitspakete	3
2.1	Arbeitspaket 1 – Projektstart	3
2.2	Arbeitspaket 2 – CI und UI-Design.....	4
2.3	Arbeitspaket 3 – RichPod-Server	4
2.4	Arbeitspaket 4 – RichPod-Editor.....	5
2.5	Arbeitspaket 5 – RichPod-Player.....	5
2.6	Arbeitspaket 6 – Podcast-Episoden	6
2.7	Arbeitspaket 7 – Dissemination	7
2.8	Arbeitspaket 8 – Projektabschluss.....	7
3	Zusammenfassung Planaktualisierung.....	7
4	Öffentlichkeitsarbeit/ Vernetzung.....	8
4.1	Blogbeiträge.....	8
5	Eigene Projektwebsite.....	8

1 Einleitung

Es zeigte sich, dass gerade die Produktentwicklung heute sehr stark userzentriert arbeitet. Zum Zeitpunkt des Antrags war nur eine grobe Idee für ein Produkt vorhanden, aber wir kannten noch nicht die eigentlichen Nutzer*innenbedürfnisse. Darum haben wir uns mehr Zeit zum Projektstart genommen und genauer auf die Bedürfnisse von Podcast-Hosts geachtet. Ohne diese Erkenntnisse entwickelt man möglicherweise in die falsche Richtung oder übersieht kritische Usability-Aspekte, die das Produkt später in seiner Entfaltung aufhalten.

Darum haben wir auch im Mai den Zeitplan der Arbeitspakete adaptiert: Zuerst den Player entwickeln, damit wir schneller User-Feedback erhalten und erst danach die Backend-Komponenten und den Editor. Schließlich definiert auch der Player mit seinen Funktionen schon viel für den Podcast-Erstellenden und natürlich auch für die Hörer*innenschaft, nicht der Editor oder das Daten-Backend. Zusätzlich dazu ist der Podcast-Markt hart umkämpft mit etablierten Playern wie Spotify oder Apple Podcasts, die auch für die Erstellung harte Vorgaben etablieren und so ihre eigenen Plattformen stärken. Ohne Prototyp fehlt die Klarheit darüber, welches Alleinstellungsmerkmal umsetzbar ist.

2 Status der Arbeitspakete

2.1 Arbeitspaket 1 – Projektstart

Es gab keine größeren Auffälligkeiten in diesem Arbeitspaket. Die Projektwebseite samt Domain wurde angelegt und alle Projektdaten auf der Netidee-Seite initial befüllt. Außerdem wurden bisher vier Blogbeiträge für die Webseite der Netidee erstellt.

Vorm eigentlichen Umsetzungsbeginn haben wir noch einmal die aktuellen Podcast-Plattformen und das Umfeld unseres Projekts genauer betrachtet. Folgende Aspekte standen dabei im Vordergrund:

- Distribution von Podcasts
- Monetarisierung der Inhalte
- Gegensteuern der fortschreitenden Zentralisierung von Podcast-Infrastruktur
- Bewerbung und Marketing für Podcaster*innen

Bei dieser Umfeldanalyse bemerkten wir, dass wir unsere Arbeitspakete anders priorisieren müssen: Zuerst einen Player, der mit Prototyp-Daten bzw. mit Dummy-Daten arbeitet und erst danach den Editor und finalen Server. Deshalb wurde AP 5 vorgezogen und direkt nach AP 2 begonnen. Diese Änderung wurde auch der Netidee am 12. Mai per Mail mitgeteilt.

2.2 Arbeitspaket 2 – CI und UI-Design

Für die Erstellung einer Basis-CI und eines UI-Entwurfs wurde Lisa Vietze extern beauftragt. Sie hatte bereits aus einem früheren Projekt Erfahrung mit der Netidee und kennt daher die grundsätzlichen Voraussetzungen.

Nach einigen Iterationen wurde eine erste Landing Page auf <https://richpods.org/> bereitgestellt. Diese wurde noch um eine Team- und Kontakt-Seite erweitert. Bis zum öffentlichen Start des Players wird sie noch um Kleinigkeiten und Projektinformationen erweitert.

Bei der Gestaltung haben wir besonders auf Barrierefreiheit nach WCAG 2.2 Level AA geachtet. Durch das Inkrafttreten des Barrierefreiheitsgesetzes (BaFG) und dem European Accessibility Act liegt nämlich besonderer Fokus auf diesem Thema. Dazu wurde auch ein eigener Blogeintrag im Netidee-Blog verfasst:

<https://www.netidee.at/richpodsorg/barrierefreiheit-gehoert-dazu>

2.3 Arbeitspaket 3 – RichPod-Server

Für dieses Arbeitspaket wurden vor allem grundlegende Vorarbeiten vorgenommen, z.B. die Technologiewahl getroffen. Wir werden die Schnittstelle mit GraphQL und Node in TypeScript implementieren und auf der Google Cloud Plattform hosten.

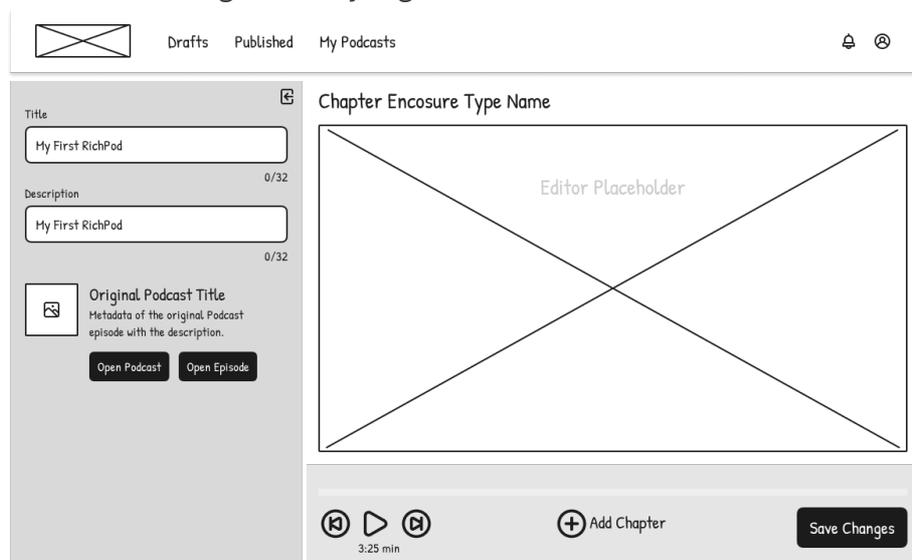
Da die Erstellung des Players (vorgezogenes AP 5) ein prototypischer Server sinnvoll war, sind bereits sehr viele grundlegende Aufgaben abgeschlossen und die einzelnen Teilebereiche des Servers so weit entworfen, dass eine weitere Umsetzung kein größeres Problem darstellen sollte. Noch offen ist die künftige Speicherung der Daten, wobei hier Postgres ins Auge gefasst wurde. Außerdem gilt es noch zu evaluieren, wie das Login-System funktionieren soll.

Aus Usersicht braucht es noch eine finale Entscheidung, ob es zuerst eine Verifikation als „Podcast-Owner“ braucht, bevor jemand einen RichPod-Podcast für einen bestehenden

Podcast anlegen kann. Wir bevorzugen im Moment eine offene Lösung, wo nur eine nachträgliche Verifikation vorgenommen wird (ähnlich zu populären Social Media Plattformen mit „Blue Badges“).

2.4 Arbeitspaket 4 – RichPod-Editor

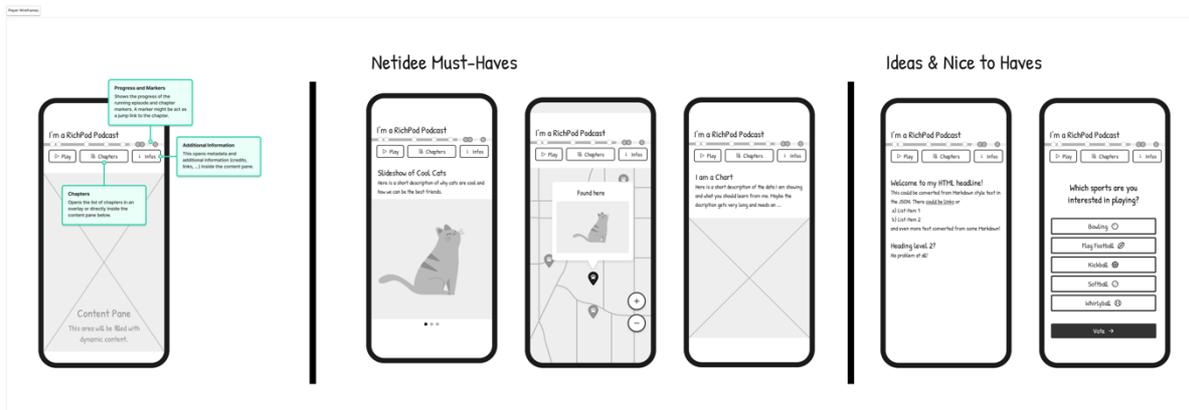
Für den Editor wurde ein erster Wireframe-Entwurf erstellt. Dabei ist aufgefallen, dass einige UI-Komponenten zwischen Player und Editor geteilt werden müssen. Darauf wurde in der weiteren Entwicklung vom Player geachtet.



Die Hauptarbeit an diesem Arbeitspaket wird im Ende Q3 und wesentlich in Q4/2025 erfolgen. Da wir mehr Ressourcen in den Player (AP 5) stecken wollen als ursprünglich geplant, werden einige Teile vom Editor nur als Rumpf bzw. nur als MVP-Variante entwickelt werden.

2.5 Arbeitspaket 5 – RichPod-Player

In einem ersten Schritt wurden Basis-Wireframes erstellt. Grundlage bildete eine Umschau unter diversen Podcast-Playern am Markt, u.a. Spotify, Apple Podcasts, Overcast, Audible, Google Podcasts, Castbox und weitere Web-basierte Player.



Diese Wireframes bildeten die Grundlage für die weitere Umsetzung.

Eine unerwartete Herausforderung war die Synchronisation einzelner UI-Komponenten mit dem jeweiligen Abspielzustand der Podcast-Audiodatei. Die Architektur der Frontend-Applikation musste hierfür nachträglich noch einmal adaptiert werden, damit alle Audio-Events und -Zustände auch korrekt in den UI-Komponenten aufscheinen.



Ein weiterer Mehraufwand entstand durch die Anforderung, dass ein RichPod auch auf einer beliebigen Webseite integriert werden soll können. Dafür war ebenfalls eine Adaptierung der Grundarchitektur vom Frontend notwendig, die aber recht zügig umgesetzt werden konnte.

Basierend auf dem Grundlayout aus AP 2 wurde im August noch eine leichte Adaptierung am UI vorgenommen. Wir haben gemerkt, dass unsere Komponenten zwar gut auf Mobilgeräten funktionieren, aber wesentliche Schwächen im Desktop-View hatten. Eine stärkere Beachtung von Desktop-Ansichten schon im Designprozess hätte hier den leichten Mehraufwand verhindert. Die Integration des Design-Updates dauert zum aktuellen Zeitpunkt noch an.

2.6 Arbeitspaket 6 – Podcast-Episoden

Durch die Vorverlegung der Wien-Wahl ist die Idee, einzelne Kandidat*innen-Interviews im Podcast als RichPod aufzubereiten, leider abhandengekommen. Allerdings hat Sonja Harter bei allen Vorproduktionen von Episoden entsprechendes Material für RichPod-Elemente gesammelt. Dieser Prozess hat auch wertvolle Inputs in den weiteren

Entwicklungsprozess vom Player gegeben, da uns erst so klarer wurde, welche Anforderungen dann in der Realität auftreten.

In den ersten Projektmonaten waren jedoch regelmäßiges Brainstorming viel wesentlicher für den Projektfortschritt. In der Entwicklung des Players wurden immer wieder Feedback-Loops eingebaut. In agilen Methoden ausgedrückt, trat Sonja Harter hier als Product Owner auf, für die das Produkt schlussendlich gebaut wird.

Gerade in diesem Punkt war es zum Zeitpunkt der Antragsstellung schwer, die einzelnen Maßnahmen und zusätzlichen Aktivitäten festzulegen. Wir werden daher im Q4 und Q1/2026 hier noch einiges zu tun haben.

2.7 Arbeitspaket 7 – Dissemination

Keine Tätigkeiten.

2.8 Arbeitspaket 8 – Projektabschluss

Keine Tätigkeiten.

3 Zusammenfassung Planaktualisierung

Im ersten Projektplan wurde bereits angemerkt, dass die Entwicklung der Arbeitspakete Player (AP 5), Server (AP 3) und Editor (AP 4) parallel erfolgen könnte bzw. es keine harten Abhängigkeiten zwischen diesen drei Komponenten gibt. Aus heutiger Sicht ergeben sich folgende Enddaten der einzelnen Arbeitspakete:

Arbeitspaket	Plan	Ist / Plan Neu
AP 1 – Projektstart	03.12.2024	03.12.2024
AP 2 – CI und UI	21.03.2025	18.04.2025
AP 3 – Server	28.02.2025	23.12.2025
AP 4 – Editor	29.08.2025	23.01.2026
AP 5 – Player	07.11.2025	07.11.2025
AP 6 – Podcast-Episoden	31.01.2026	31.01.2026
AP 7 – Dissemination	31.01.2026	31.01.2026
AP 8 – Projektabschluss	27.02.2026	27.02.2026

→ **Wir liegen – trotz Umstellungen im Arbeitsablauf – weiter im Gesamtzeitplan.**

4 Öffentlichkeitsarbeit/ Vernetzung

Beschreibung der bereits erfolgten Öffentlichkeitsarbeit oder Vernetzung, bzw. Beschreibung des Plans künftiger Aktivitäten

Anfang Oktober nehmen wir bei der Konferenz „re:think media“ im Museumsquartier teil. Dort werden wir auch unseren Player erstmals in der Branche breit herzeigen können. Wir haben uns auch über eine Teilnahme beim EBU (European Broadcasting Union) Podcast Forum informiert, da dies eine potentielle Vernetzungsmöglichkeit gewesen wäre. Allerdings ist eine Teilnahme – trotz inhaltlich großen Übereinstimmungen – außerhalb der EBU nicht möglich gewesen. Dennoch werden wir versuchen, hier weiter in Kontakt zu bleiben und RichPods in einem reiferen Projektstatus vorzustellen.

Außerdem wurde bereits Kontakt mit bekannten österreichischen Podcaster*innen (Schwerpunkt News / Politik) aufgenommen. Wir möchten anhand vom ersten Player-Prototypen dann im Q3 und Q4 hier weiteres Feedback einholen und unser Produkt weiterentwickeln.

4.1 Blogbeiträge

Folgende Blogbeiträge wurden erstellt:

- [RichPods startet: Podcasts interaktiv gedacht](#)
- [Podcasting und Plattformen – ein Überblick](#)
- [Wie Personas helfen nutzerzentriert zu gestalten](#)
- [Barrierefreiheit gehört dazu](#)

5 Eigene Projektwebsite

Wir haben eine erste Landing Page erstellt, die grundlegende Informationen zum Projekt enthält und zumindest das Team und einen Hinweis auf die Netidee-Blogbeiträge aufweist.

<https://richpods.org/>

Das Github-Repository ist vorerst noch privat, wir planen für ~ Mitte September eine erste öffentliche Version vom Source Code zu veröffentlichen.