

FRAUNHOFER-INSTITUT FÜR OFFENE KOMMUNIKATIONSSYSTEME FOKUS

Projektabschlussbericht KiezRadar

ortsbezogen – personalisiert – proaktiv

Susanna Kuper, Milko Monecke, Isabell von Falkenhausen, Lutz Nentwig, Ekkart Kleinod, Susann Pocha, Tim Zimmermann



gefördert durch



Inhalt

1 Einleitung

2 AP 1: UX-Konzept

- 2.1 Begriffsdefinition
 - 2.1.1 Was bedeutet Usability?
 - 2.1.2 Was ist der Unterschied zur User Experience?
 - 2.1.3 Was ist Human-centered Design?
- 2.2 Normen und Standards
 - 2.2.1 DIN ISO Norm 9241
 - 2.2.2 Vorgaben zur digitalen Barrierefreiheit
- 2.3 Usability-Anforderungen an mobile Apps
- 2.4 Nutzerzentrierung im Projekt KiezRadar
 - 2.4.1 Human-centered Design-Methoden
 - 2.4.2 Visuelles Design
 - 2.4.3 Kommunikation und Transparenz
 - 2.4.4 Informationsarchitektur & Navigation
 - 2.4.5 Interaktionsdesign

3 AP 1: Anforderungserhebung

- 3.1 Bürger:innen-Workshop
 - 3.1.1 Vorbereitung
 - 3.1.2 Konzeption und Durchführung
 - 3.1.3 Ergebnisse und Auswertung
- 3.2 Workshop mit Verwaltungsmitarbeiter:innen
 - 3.2.1 Vorbereitung
 - 3.2.2 Konzeption und Durchführung
 - 3.2.3 Ergebnisse und Fazit
- 3.3 Einbindung von Akteuren aus der Wirtschaft
- 3.4 Personas
 - 3.4.1 Methodik
 - 3.4.2 Umsetzung
 - 3.4.3 Ergebnisse

4 AP 1: Backlog und Anwendungsszenarien

- 4.1 User Journey
 - 4.1.1 Prozessschritte der Nutzung
 - 4.1.2 Supportfunktionen
- 4.2 Prozessschritte der Nutzung
 - 4.2.1 Verfügbarkeit und Installation
 - 4.2.2 Informationsmenge steuern
 - 4.2.3 Informationen zu Teilnehmungsformaten und Veranstaltungen
 - 4.2.4 Kartenfunktion und standortbezogene Nutzung
 - 4.2.5 Beteiligungen wahrnehmen
 - 4.2.6 Projektergebnisse verfolgen
 - 4.2.7 Interoperabilität/ Integration

- 4.2.8 Kommunikation
- 4.3 Supportfunktionen
 - 4.3.1 Hilfe und Datenschutz
 - 4.3.2 Einstellungsmöglichkeiten
 - 4.3.3 Barrierefreiheit
 - 4.3.4 Navigation und Bedienung (Usability)
- 4.4 Anwendungsszenarien
 - 4.4.1 Beispiel Quartiersmanagement
 - 4.4.2 Beispiel mein.berlin.de
 - 4.4.3 Beispiel Ratsinformationssystem
 - 4.4.4 Beispiel Projekte und Veranstaltungen entlang einer Route

5 AP 2: Konzept der Inhaltserstellung

- 5.1 Nutzung öffentlich verfügbarer Quellen
- 5.2 Erstellung eigener Inhalte
- 5.3 Nutzung öffentlicher Quellen und Anreicherung der Inhalte
- 5.4 Inhaltserstellung im Prototyp

6 AP 3: Technisches Konzept, technische Architektur

- 6.1 Architekturprinzipien
 - 6.1.1 Entkopplung der Quelldaten von den KiezRadar-Daten
 - 6.1.2 Eigene Datenhaltung und eigenes Datenformat für KiezRadar
 - 6.1.3 Einheitliche KiezRadar-API
- 6.2 KiezRadar-API
- 6.3 Datenaufbereitung
- 6.4 Frontend

7 AP 4: Entwicklung

- 7.1 Plattformen und Docker
- 7.2 Datenimport und -aufbereitung
- 7.3 Backend
- 7.4 Frontend
 - 7.4.1 Design und Funktionskonzept
 - 7.4.2 Android App Implementierung
 - 7.4.3 Abweichungen vom Architekturkonzept

8 AP 5: Test- und Pilotphase

- 8.1 Planung
- 8.2 Ablauf
- 8.3 Ergebnis

9 Weiteres

- 9.1 Zusammenarbeit
- 9.2 Mediale Präsenz
- 9.3 Open Source

10 Lessons Learned

- 10.1 Nutzerzentrierter Entwicklungsansatz
- 10.2 Einbindung der Verwaltung
- 10.3 Tests und Feedback
- 10.4 Qualität der Datenquellen
- 10.5 Open Source
- 10.6 Hybride vs. native App-Entwicklung
- 10.7 Die Idee kommt an – Bedarf für die KiezRadar-App vorhanden

11 Weiterentwicklung/Ausblick

- 11.1 Zukunft Backend
 - 11.1.1 Was fehlt?
 - 11.1.2 Optionen für die Zukunft
 - 11.1.3 Was bringt das?
- 11.2 Zukunft Datenaufbereitung
 - 11.2.1 Was fehlt?
 - 11.2.2 Optionen für die Zukunft
 - 11.2.3 Was bringt das?
- 11.3 Zukunft App
 - 11.3.1 Was fehlt?
 - 11.3.2 Optionen für die Zukunft
 - 11.3.3 Was bringt das?

12 Glossar, Abkürzungen

1 Einleitung

Viele Bürger:innen wollen sich an der Gestaltung ihrer Stadt beteiligen – und das zielorientiert, unkompliziert und direkt. Damit verbunden sind neue Anforderungen an das Zusammenspiel von Politik, Verwaltung und Bürgergesellschaft. Die Erwartungen sind dabei hoch: Bessere Qualität der Planung, höherer Gemeinsinn und mehr Digitalisierung.

Doch wie erfahren die Bürger:innen von aktuellen Planungen und Entscheidungen? Wie bekommen Bürger:innen relevante Angebote in Ihrem Kiez mit? Wie werden Bürger:innen zeitnah und ortsbezogen informiert, ohne dass sie mit Informationen überschwemmt werden? Die derzeitige Vielfalt der vorhandenen digitalen Informationskanäle mit unterschiedlicher Zielsetzung und verschiedenen Zielgruppen erschwert den Überblick und die gezielte Informationssuche interessierter Bürger:innen.

Damit Bürger:innen sich nicht selbst durch die digitalen Angebote Berlins wühlen müssen, soll die KiezRadar-App proaktiv, übersichtlich und bedarfsgerecht über wichtige Ereignisse aus Politik und Verwaltung informieren. Die Bürger:innen können selbst auswählen, zu welchen Themen und mit welchem Ortsbezug sie benachrichtigt werden möchten. Hierfür bündelt KiezRadar verschiedene Informationen aus bestehenden Plattformen und bereitet sie adressatengerecht auf. Als Datenquelle stehen die Plattform mein.berlin.de und die Ratsinformationssysteme der Bezirke Berlins im Fokus des Projekts. Aber auch die Vernetzung mit Informationsseiten z. B. lokaler sozialer Netze oder der Wirtschaft sollen mitgedacht werden. Die Übernahme und Zusammenführung der Daten sollen dabei automatisiert durch die Nutzung vorhandener Schnittstellen erfolgen.

Auch das Projekt KiezRadar selbst wird dabei als Beteiligungsprojekt verstanden. Zusammen mit dem neuen digitalen Experimentierlabor für die Stadt der Zukunft, dem CityLAB Berlin, sollen neue Konzepte der Bürger-Verwaltungs-Kommunikation erprobt werden.

Das Projekt KiezRadar folgt einem agilen und nutzerzentrierten Ansatz, bei dem Human-centred Design-Methoden eingesetzt werden. Die Stoßrichtung von KiezRadar steht zwar fest – doch wie die App gestaltet werden soll, welche Bedienkonzepte die Nutzer:innen favorisieren und welche wichtigen Ereignisse die Bürger:innen in Berlin mitbekommen möchten – all das soll in einem offenen partizipativen Prozess, zusammen mit den zentralen Stakeholdern aus der Bürgergesellschaft, Verwaltung und Wirtschaft herausgefunden werden.

2 AP 1: UX-Konzept

Ein wesentliches Ziel des Projekts KiezRadar ist es, aufzuzeigen, wie ein Entwicklungsprozess der öffentlichen Verwaltung möglichst nutzerzentriert erfolgen kann. Dabei steht das gesamte Nutzungserlebnis bzw. die User Experience im Fokus – von der Anforderungserhebung, der Konzeption des App-Designs, der Entwicklung des Prototypens bis hin zur späteren KiezRadar-App. Die Umsetzung einer ausgereiften App-Lösung ist zwar nicht Teil des Projektes, wird aber durch die Aufnahme der Anforderungen und der Evaluation des Prototypens durch Nutzertests unterstützt.

Für eine nachhaltige und ausgereifte User Experience (UX) ist ein planvolles Vorgehen sinnvoll, bei dem die Nutzer:innen systematisch in den Mittelpunkt gerückt werden. Das nachfolgende UX-Konzept beschreibt die Bausteine, Leitlinien und eingesetzten Instrumenten bzw. Methoden, die im Designprozess der KiezRadar-App eine optimale User Experience sicherstellen sollen.

Da User-Experience-Design ein multidisziplinäres Feld ist, vereint das Konzept Komponenten und Erkenntnisse aus verschiedenen Bereichen, wie Design, Mensch-Computer-Interaktion, Psychologie oder Software-Entwicklung. Gegenstand des UX-Konzeptes sind u.a. das visuelle Design, Informationsarchitektur und Navigation, Interaktionsdesign und Kommunikationsprozesse. Zudem wird auch auf die Besonderheiten mobiler Anwendungen eingegangen und daraus resultierenden Anforderungen.

Zuvor erfolgt eine allgemeine Einführung in die Thematik durch die Definition und Abgrenzung der Begriffe User Experience und Usability sowie eine Vorstellung der wichtigsten Standards und Normen.

2.1 Begriffsdefinition

2.1.1 Was bedeutet Usability?

Die Gebrauchstauglichkeit (engl. Usability) ist »das Ausmaß, in dem ein System wie bspw. eine digitale Anwendung oder eine Webseite durch bestimmte Nutzer:innen genutzt werden kann, um in einem bestimmten Nutzungskontext bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen.«¹

- Effektivität: Kann das gewünschte Ziel mit dem System genau und vollständig erreicht werden?
- Effizienz: Wie ist das Verhältnis der gewünschten Ziele und dazu eingesetzten Ressourcen wie etwa Zeit, kognitive Leistung, Geld oder Material definiert?
- Zufriedenheit: Ist die Wahrnehmung und Reaktion der Nutzer:innen, die aus der Benutzung des Systems resultieren, positiv?

Der Grad an Gebrauchstauglichkeit wirkt sich demnach auf die Qualität und Effizienz der Arbeit, die Gesundheit und Motivation der Mitarbeiter:innen sowie auf die Akzeptanz und Nachhaltigkeit von IT-Projekten aus.

¹ Vgl. DIN EN ISO 9241-11 (1998) In der deutschen Fassung dieser Norm wird Usability mit dem Terminus »Gebrauchstauglichkeit« gleichgesetzt. Weitere Definitionsversuche sind u. a. bei folgenden Autoren zu finden: Nielsen (1993); Dumas und Redish (1999);

2.1.2 Was ist der Unterschied zur User Experience?

Unter »User Experience« werden Wahrnehmungen und Reaktionen von Nutzer:innen verstanden, die aus der tatsächlichen und/oder der erwarteten Benutzung eines Systems resultieren. Das Nutzererlebnis konzentriert sich auf die Vorlieben, Wahrnehmungen, Einstellungen, Emotionen sowie die physischen und psychologischen Reaktionen, die sich vor, während und nach der Nutzung ergeben.

Es wird demnach eine breiter aufgestellte Perspektive eingenommen, die das Nutzererlebnis über den gesamten Lebenszyklus eines Produktes erfasst – von den ersten Berührungspunkten des Nutzers mit dem Produkt durch Werbung, über das Käuferlebnis und die Inbetriebnahme bis hin zur Weiterempfehlung oder dem Umstieg auf ein Konkurrenzprodukt. Gebrauchstauglichkeit ist darunter eine Variable, die einen Beitrag zum Nutzererlebnis des Systems erbringt.² Während sich die Gebrauchstauglichkeit üblicherweise mit Zielen befasst, die von einer Benutzergruppe geteilt werden, behandelt das Nutzererlebnis individuelle Ziele, die persönliche Motivationen einschließen können, inklusive der Bedürfnisse, neue Kenntnisse und Fähigkeiten zu erwerben, die persönliche Identität zu übermitteln und angenehme Erinnerungen hervorzurufen.³ Die folgende Abbildung illustriert die Schnittmenge von Usability und User Experience.

Abbildung 1: Abgrenzung Usability und User Experience.



2.1.3 Was ist Human-centered Design?

»Human-centered Design« (HCD) oder auch »User-centered Design« entspringt dem Gedanken, die Ära schlecht designter Softwareprodukte durch die Ausrichtung auf die Nutzerbedürfnisse sowie Berücksichtigung einschlägiger Usability- und Designrichtlinien hinter sich zu lassen. HCD umfasst sowohl die Ausrichtung an bestimmten Prinzipien, als auch konkrete Vorgehensmodelle oder eine Sammlung möglicher Methoden. Mit der DIN EN ISO 9241-210 existiert ein maßgebendes Dokument für den HCD-Prozess. Einen wesentlichen Teil der Norm bildet ein aus vier Designaktivitäten bestehendes Framework, bei dem das Vorgehen des nutzerorientierten Gestaltungsprozesses spezifiziert wird (DIN EN ISO 9241-210 (2008), S.10). Die vier Aktivitäten lauten wie folgt:

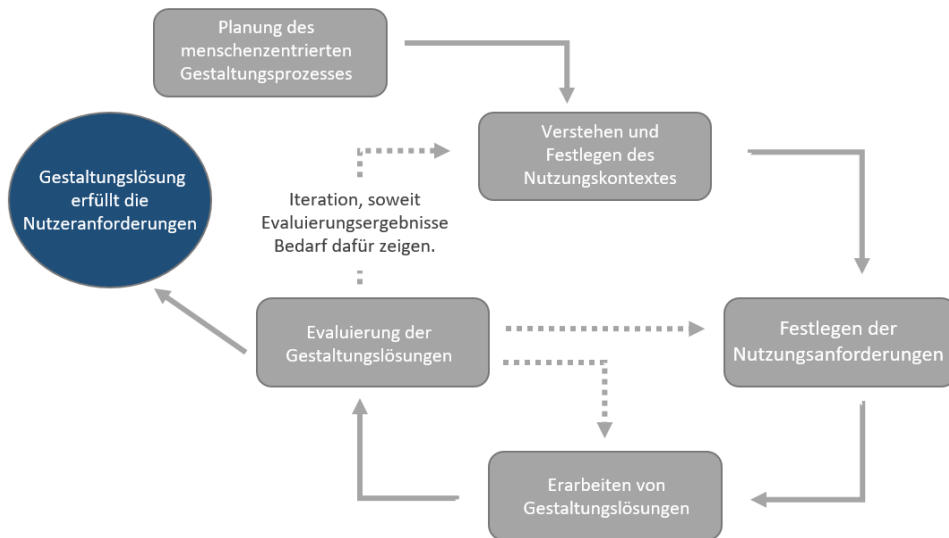
- Nutzungskontext verstehen und beschreiben

² Vgl. Fischer et al. (2013).

³ ISO Norm 9241 Teil 11 (2006).

- Nutzungsanforderungen spezifizieren
- Gestaltungslösungen entwickeln
- Gestaltungslösungen aus der Benutzerperspektive evaluieren

Abbildung 2: HCD-Prozess nach DIN EN ISO 9241-210.



2.2 Normen und Standards

Nachfolgend werden die wichtigsten Standards in Bezug auf die Gebrauchstauglichkeit digitaler Anwendungen vorgestellt, die im Zuge des Designprozesses als Vorgabe für das App-Design angewendet werden.

2.2.1 DIN ISO Norm 9241

Einen Maßstab für die softwareergonomische Gestaltung bilden die Normen Teil 11 bis 17 sowie 110 der Normenreihe EN ISO 9241 »Ergonomie der Mensch-System-Interaktion«, die in den Jahren 1996 bis 1999 verabschiedet und anschließend als DIN-Normen übernommen worden sind. In Teil 110 Grundsätze der Dialoggestaltung werden Qualitätskriterien definiert, die bei der Anwendung der Evaluationsmethoden berücksichtigt werden sollten: Aufgabenangemessenheit, Selbstbeschreibungsfähigkeit, Erwartungskonformität, Lernförderlichkeit, Steuerbarkeit, Fehlertoleranz sowie Individualisierbarkeit.

2.2.2 Vorgaben zur digitalen Barrierefreiheit

Barrierefreies Design setzt das Ziel, digitale Angebote für alle Menschen mit Behinderung uneingeschränkt und selbstständig nutzbar zu gestalten. Digitale Angebote wie bspw. mobile Anwendungen müssen für Menschen mit Beeinträchtigungen wahrnehmbar, verständlich, bedienbar und robust designet werden. Mögliche Beeinträchtigungen können dabei u.a. Blindheit und Sehbehinderung, Gehörlosigkeit und nachlassendes Hörvermögen, Lernbehinderungen, kognitive Einschränkungen, eingeschränkte Bewegungsfähigkeit, Sprachbehinderungen, Photosensibilität und Kombinationen aus diesen Behinderungen darstellen. Ein barrierefreies Design wirkt sich überdies positiv auf die Benutzerfreundlichkeit aus.

Die EU-Richtlinie 2016/2102 ist an alle Mitgliedstaaten der EU mit dem Ziel gerichtet, die Anforderungen der Barrierefreiheit auf Webseiten und mobilen Apps europaweit zu vereinheitlichen. Die Umsetzung in nationales Recht ist in Berlin mit dem Barrierefreie-Informationstechnik- und Kommunikationstechnik-Gesetz Berlin (BIKTG Bln, 2019) geschehen. Aufgrund des Gesetzes zur Förderung des E-Government (E-Government-Gesetz Berlin – EGovG Bln) und des Gesetzes über die barrierefreie Informations- und Kommunikationstechnik Berlin (BIKTG Bln) ist die Berliner Verwaltung verpflichtet, sich bei allen IT-unterstützten Aufgaben nach folgenden rechtlichen Bestimmungen zu richten:

- Barrierefreie – Informationstechnik – Verordnung – BITV
- Web Content Accessibility Guidelines – WCAG
- Accessibility requirements suitable for public procurement of ICT products and services in Europe – EN 301 549

Die Kompetenzstelle für digitale Barrierefreiheit und Usability definiert folgende Anforderungen an die Webseiten und webbasierte Software⁴:

- Alle Bedingungen der BITV oder gegebenenfalls der Konformitätsstufe AA beim WCAG müssen erfüllt sein. Wenn sie nicht beim Abschluss des Vertrages bzw. der Softwareeinführung erfüllt sind, müssen sie nach Ablauf eines Jahres erfüllt sein.
- Die Berliner Ausschlusskriterien müssen bei Abschluss des Vertrages bzw. bei Einführung der Software erfüllt sein.
- Die Webseite bzw. Software muss durch einen Screenreader vollständig nutzbar sein.
- Alle Dokumente, die auf der Webseite verwendet werden, müssen barrierefrei sein und den Berliner Standards zur Barrierefreiheit entsprechen.
- Erstellte Software Dokumente, müssen diese barrierefrei sein und den Berliner Standards zur Barrierefreiheit entsprechen.

2.3 Usability-Anforderungen an mobile Apps

Die Nutzung von mobilen Anwendungen ist gegenüber Webanwendungen auf traditionellen Desktop-Computern mit besonderen Rahmenbedingungen verbunden aus denen sich andere Design-Anforderungen ergeben. Es gibt physische Unterschiede, z. B. in Bezug auf die Größe (Bildschirm, Bedienelemente), den Arbeitsspeicher, die Bedienung (Touch-Gesten) oder Unterschiede im Nutzungskontext (Nutzung unterwegs), woraus sich verschiedene Herausforderungen ergeben. Die geringere Bildschirmgröße limitiert die Fläche zur Darstellung von Inhalten. Durch die Nutzung an wechselnden Orten muss mit verschiedenen Umweltbedingungen umgegangen werden. Umgebungsgeräusche können sich bspw. auf die Konzentrationsfähigkeit der Nutzer:innen auswirken, schlechte Lichtverhältnisse die Lesbarkeit beeinflussen oder technischen Voraussetzungen, wie schwache oder gar keine Netzwerkleistung die Nutzung von Anwendungen beschränken. Die besonderen Rahmenbedingungen für die Nutzung mobiler Apps sollten entsprechend bei der Konzeption mitgedacht werden.

Aufgrund des limitierten Platzes für Inhalte sollten weniger Inhalte auf einmal dargestellt werden, als bei einer Anwendung für Desktop-Computer. Damit Inhalte nicht verloren

⁴ <https://www.berlin.de/sen/inneres/moderne-verwaltung/digitale-barrierefreiheit/berliner-standards/artikel.807556.php#checkliste>

gehen, liegt bei mobilen Anwendungen ein größerer Fokus darauf, die Nutzer:innen bei der Inhaltserschließung zu leiten. Dabei werden Inhalte oder Funktionen eher Schritt für Schritt erschlossen, anstatt sie auf einmal zu präsentieren. Möglichkeiten dies zu strukturieren bietet u.a. die Nutzung von Wizards, Reitern, Listen oder aufklappbaren Inhaltsbereichen. Auch Beschriftungen von Bedienelementen werden bei mobilen Anwendungen sparsam eingesetzt. Dabei ist es von großer Bedeutung, dass die Grafiken, die für die Bedienelemente eingesetzt werden, erwartungskonform und selbsterklärend sind. Durch die wechselnden Lichtverhältnisse sollte ein ausreichender Kontrast bei Grafiken und Schrift sowie eine ausreichende Größe der Schrift sichergestellt sein. Die Benutzerführung sollte auf die Bedienung durch Touch-Gesten hin optimiert werden. Dabei sollte die Benutzerführung möglichst so gestaltet werden, dass sie größtenteils mit dem Daumen bzw. einer Hand bedient werden kann. In der Navigationsstruktur ermöglicht beispielsweise die Fußleiste ein Anklicken der wichtigen Funktionen mit dem Daumen. Der weitaus geringere Arbeitsspeicher und die fehlende Stabilität im Internetzugriff macht eine geringe Arbeits- und Telefonspeicherplatzbeanspruchung der Applikation notwendig.

2.4 Nutzerzentrierung im Projekt KiezRadar

2.4.1 Human-centered Design-Methoden

Das Projekt KiezRadar umfasst den Einsatz verschiedener HCD-Methoden, die die unterschiedlichen Phasen des HCD-Prozesses umfassen. Im Sinne der Phase „Nutzungskontext verstehen“ werden die verschiedenen Nutzergruppen (Bürger:innen, Verwaltung, Wirtschaft) mit der Durchführung von kreativen Workshops in den Entwicklungsprozess eingebunden (siehe Abschnitte 3.1 und 3.2). Die erhobenen Anforderungen werden durch die Erstellung von Personas für die zentralen Nutzergruppen ergänzt (siehe Abschnitt 3.4). Bei der Phase der „Festlegung der Nutzungsanforderungen“ werden die Anforderungen in Form von User Stories festgehalten, so dass sie möglichst einer bestimmten Nutzergruppe zugeordnet und mit einer Intention oder einem Zweck verknüpft werden. Im Backlog (Abschnitt 4) wird die Gliederung der Anforderungen anhand einer User Journey (Nutzungsreise) vorgenommen, in welcher die wichtigen Phasen, die in der Interaktion mit den Nutzer:innen auftreten können, die erforderlichen Nutzeraktionen, die Handlungsalternativen und die konkreten Berührungspunkte mit der App dargestellt werden. Als konkrete Gestaltungslösung werden Mockups sowie ein Prototyp der KiezRadar-App als KlickDummy umgesetzt. Die Mockups und der Prototyp unterlaufen wiederholt Evaluationen unter Beteiligung von Nutzer:innen.

2.4.2 Visuelles Design

Bei der Konzeption des visuellen Designs wurde darauf geachtet, einen Wiedererkennungswert im Sinne einer visuellen Identität des Projekts zu erzielen und eine Repräsentationsfläche zu schaffen, die dem Nutzungskontext der KiezRadar-App angemessen ist. Als zentrales Element wurde zu Beginn des Projekts das KiezRadar-Logo entworfen (siehe Abbildung 3). Bei dem Logo wird die Symbolik des Lautsprechers, welcher ein in der politischen Bewegung verankertes Symbol ist, aufgegriffen. Ergänzend finden sich auch Radarwellen, in diesem Kontext als Schallwellen dargestellt, im Logo wieder, die einen Bezug zu dem »Radar« in KiezRadar herstellen. Durch den zeichnerischen Stil und die bunten Farben wirkt das Logo fröhlich, verspielt und dynamisch.

Abbildung 3: Projektlogo KiezRadar.



Das Logo wird stringent im Projektverlauf verwendet, bspw. bei Flyern, der Webseite oder in Präsentationen. Ebenso werden die Farben des Logos in verschiedenen Medien und Grafiken wiederverwendet. Für den Prototypen der App wurden aus Gründen der Barrierefreiheit ergänzend noch verschiedene Grau-Töne und ein dunkles Blau hinzugefügt. Das dunkle Blau und Grau dienen dabei als Hintergrund für helle Schriftfarben und Grafiken, während die hellen Grautöne oder Weiß als Hintergrund für dunkle Schriften und Grafiken geeignet sind. Für einen WCAG-konformen Farbeinsatz wurden zudem für die Erstellung der Kartenmarker weitere Farben mitaufgenommen.

Entsprechend ergeben sich die folgenden Farbpaletten für das Projekt:

Abbildung 4: Farbpalette für Grafiken und Identity.

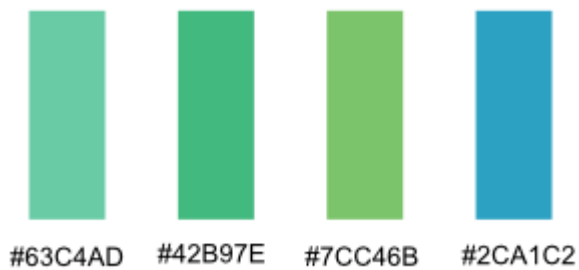


Abbildung 5: Farbpalette für Funktionen und Beschriftungen der KiezRadar-App.

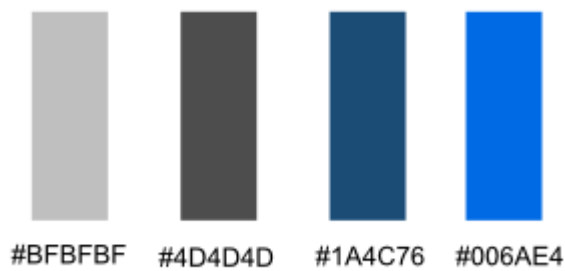
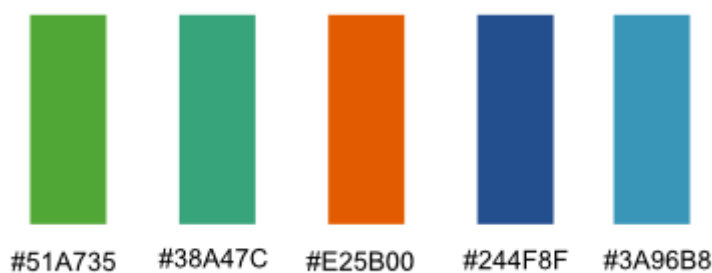


Abbildung 6: Farbpalette für die Kartenmarker.



2.4.3 Kommunikation und Transparenz

Ein wesentlicher Baustein der Transparenz und Kommunikation im Projekt bildet die Projekt Webseite von KiezRadar (<https://kiezradar.fokus.fraunhofer.de>). Auf der Webseite wird regelmäßig über den Entwicklungsstand und die Entwicklungsaktivitäten informiert. Durch das Kommunizieren von Herausforderungen, Erfolgen und Zwischenständen können Nutzer:innen die Entwicklung und bestimmte Entscheidungen im Entwicklungsprozess nachverfolgen. Zudem können die veröffentlichten Ergebnisse und Erkenntnisse für andere Projekte nachgenutzt werden. Die Webseite dient aber auch als Instrument, um Feedback einzusammeln. Nutzer:innen werden hierfür regelmäßig anhand von konkreten Fragestellungen zur Abgabe von Feedback aufgerufen.

Zur Transparenz und Nachhaltigkeit gehört auch die konsequente Offenlegung des Quellcodes der Schnittstelle, des Prototypens und der API.

2.4.4 Informationsarchitektur & Navigation

Die Informationsarchitektur ist Grundlage für die Navigation und damit eine entscheidende Komponente im App-Design. Sie umfasst die Klassifizierung und Kategorisierung aller Inhalte.

Sie bestimmt

- welche Inhalte verfügbar gemacht werden sollen,
- wie diese Inhalte benannt werden,
- wie die Inhalte strukturiert und verlinkt sein sollen und
- welche grafischen Repräsentationen genutzt werden sollen.

Inhalte sollen einfach verständlich und navigierbar sein. In mobilen Anwendungen kann dies aufgrund der geringen Bildschirmgröße eine Herausforderung sein. Die

Informationsarchitektur sollte sich deshalb auf das Wesentliche beschränken, Informationshierarchien flach halten und Menüstrukturen klar und übersichtlich abbilden.

Die Informationsarchitektur und Navigation der KiezRadar-App wurde mittels den unter Abschnitt 2.4.1 genannten nutzerzentrierten Methoden erarbeitet und validiert.

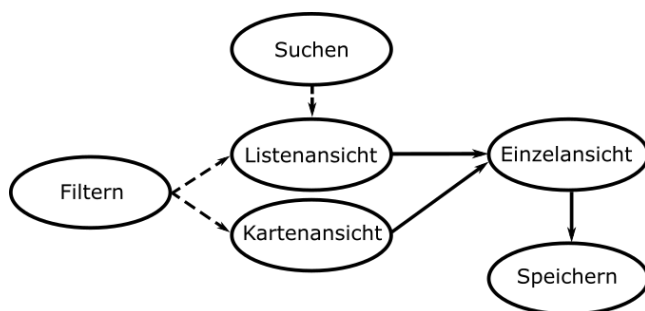
Zentraler Inhalt von KiezRadar sind die Einträge zu Veranstaltungen und Projekten sowie deren Metadaten (Kategorien, Datum, Ort). Die App sollte folgende Ansichten und Funktionen für die Einträge enthalten:

- Kartenansicht
- Listenansicht
- Einzelansicht
- Filtern von Einträgen
- Speichern von Einträgen
- Suchen von Einträgen

Die Listen- und Kartenansicht dienen der Darstellung der Einträge. Die Listenansicht bietet eine kompakte Übersicht aller Einträge und die Kartenansicht eine geographisch zugeordnete Übersicht aller Einträge. Das Filtern von Einträgen ist auf die Listen- und Kartenansicht anwendbar, so dass nur gewünschte Elemente angezeigt werden. Die beiden Ansichten sind jedoch unabhängig von dieser Funktion aufrufbar. Aus der Listen- und Kartenansicht ist das Aufrufen von Einträgen in der Einzelansicht möglich. In der Einzelansicht sind detaillierte Informationen enthalten und die Möglichkeit, den Eintrag zu speichern. Die Funktion Suche sucht in allen Einträgen und stellt das Suchergebnis in der Listenansicht dar.

Im Folgenden sind die Verbindungen zwischen Ansichten und Funktionen grafisch dargestellt:

Abbildung 7: Verbindungen zwischen Ansichten und Funktionen in der Informationsarchitektur.



Die Ansichten und Funktionen haben ähnliche Nutzungsprioritäten, wobei zu beachten ist, dass die Filterfunktion der Listen- und Kartenansicht zugeordnet ist, da nur in einer Karte oder Liste gefiltert werden kann. Als Navigationsstruktur wurde deshalb die Tabbed-View-Navigation gewählt. Die Tabbed-View-Navigation ermöglicht über eine Fußleiste am unteren Rand des Bildschirms einen schnellen und direkten Zugriff auf mehrere Funktionen der App. Zudem erleichtert sie die Navigation zwischen den Bereichen und Multitasking von verschiedenen Funktionen. Die Tabbed-View Navigation ist besonders gut für mobile Anwendungen geeignet, da durch den Daumen auf die Fußleiste zugegriffen werden kann und sie somit mit einer Hand bedienbar ist. Smartphone- und Tablet-

Nutzer:innen sind mit dieser Navigationsstruktur durch andere Anwendungen vertraut und finden sich schnell darin zurecht, da es eine gängige Navigationsstruktur ist. Die Tabbed-View-Navigation verhindert zudem eine tiefe Struktur von Inhalten, die die Navigation verlangsamen würde.

Die identifizierten Ansichten und Funktionen sind in die KiezRadar-App über die Fußleiste integriert. Diese ist in die funktionalen Bereiche Entdecken (Listenansicht), Karte (Kartenansicht), Favoriten (Speichern), Suche (Suchen) und Einstellungen unterteilt.

Die **Filterfunktion** ist in den Bereichen Entdecken und Karte anwendbar. In diesen Bereichen kann das Filtermenü über einen Button am oberen Rand des Bildschirms aufgerufen werden. Innerhalb der Entdecken-, Favoriten-, und Suchen-Bereiche werden Einträge als Liste angezeigt, um Übersichtlichkeit zu schaffen.

Abbildung 8: Erste Entwurfsphase mit zwei alternativen Mockups für die Filterfunktion.

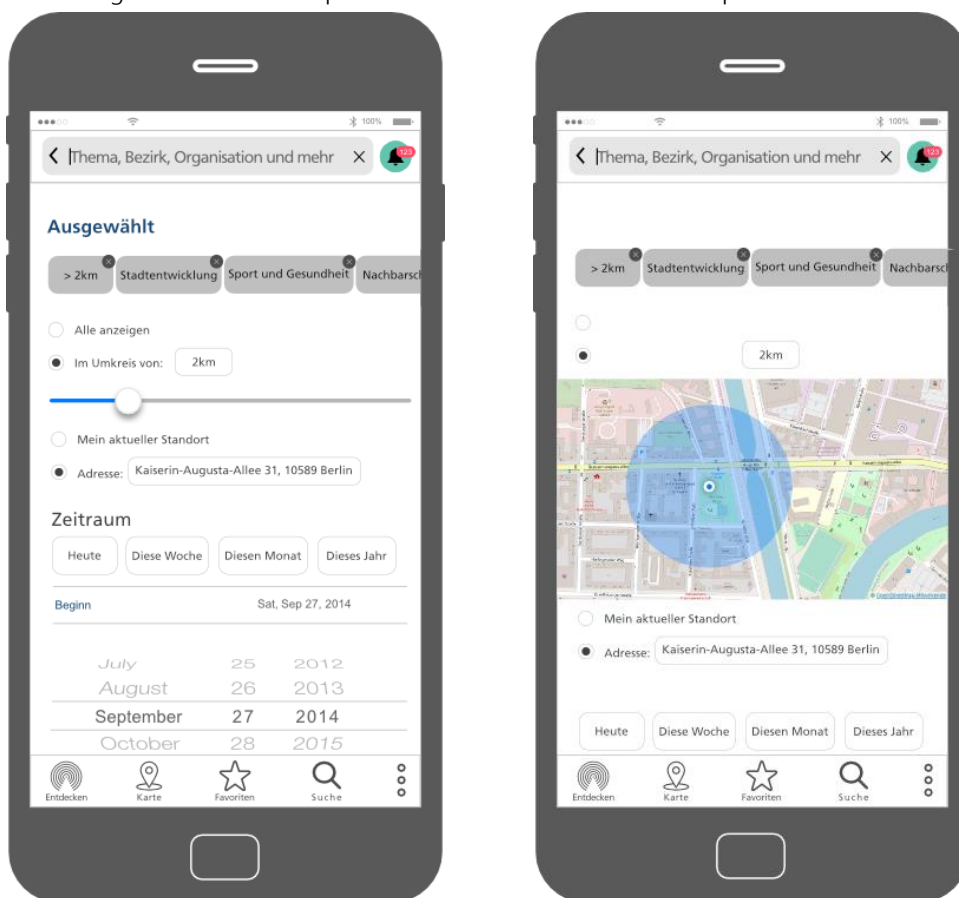
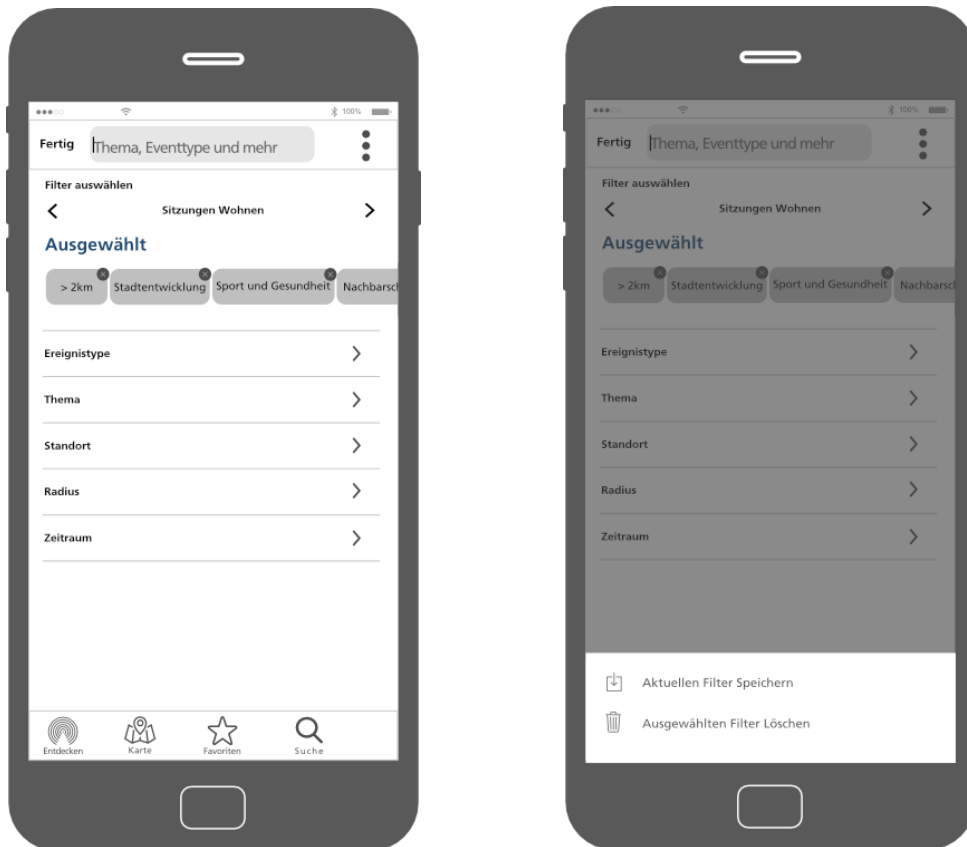
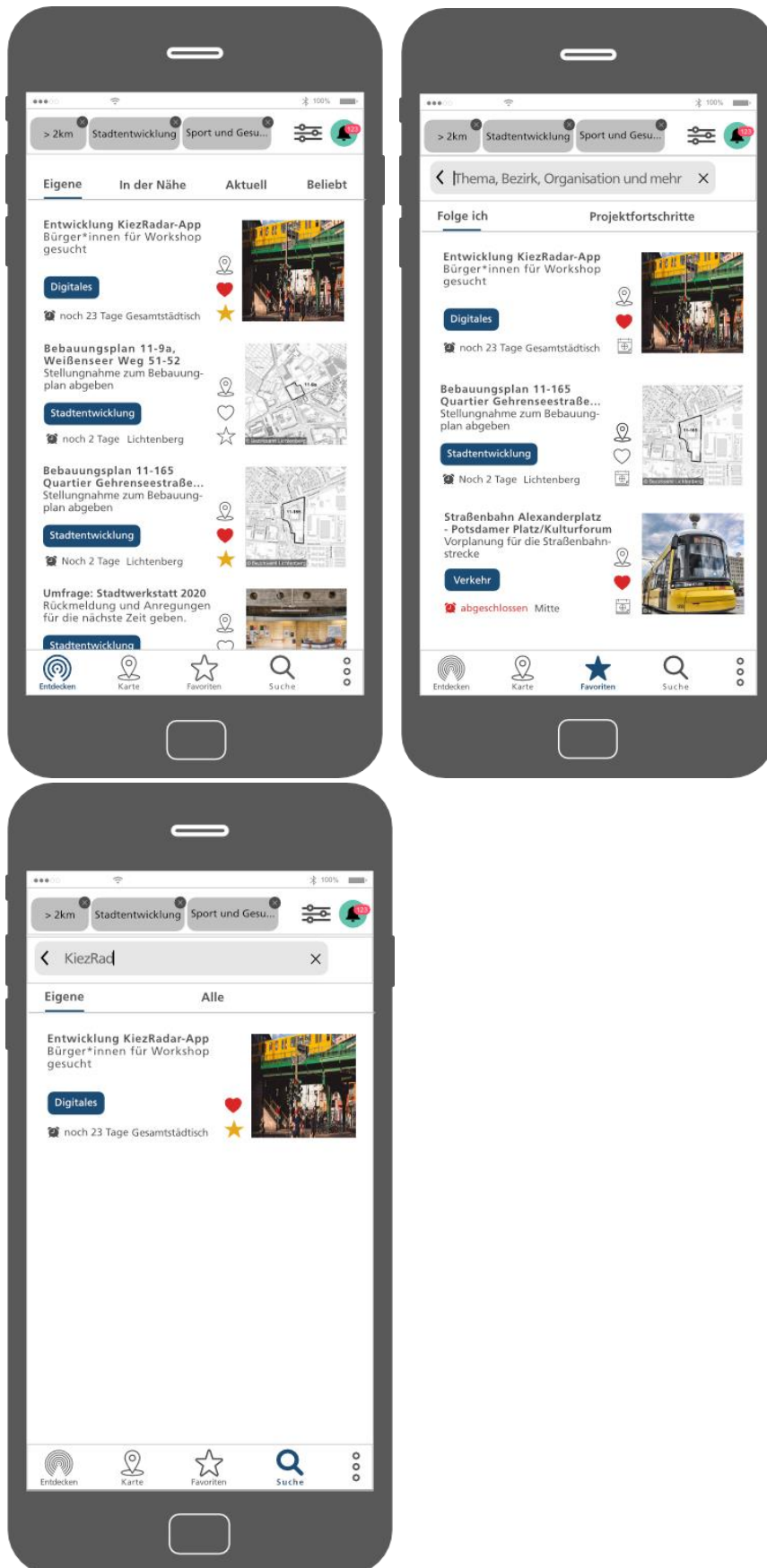


Abbildung 9: Zweite Entwurfsphase Mockups für die Filterfunktion.



Die **Listenansicht** enthält die wichtigsten Informationen zu den Einträgen wie Titel und Ort sowie ein Vorschaubild für ein visuell ansprechendes Design. Die Listenansicht verfügt in allen Bereichen (Speichern, Filtern, etc.) über das gleiche Design, um Konsistenz zu schaffen und um die Navigation für Nutzer:innen einfacher zu machen.

Abbildung 10: Mockups zu den Listenansichten (Entdecken-Funktion, Favoritenliste und Suchen).



Die **Karte** verfügt über die klassischen Kartennavigationselemente: Zoomen funktioniert durch Pinching⁵ oder »plus« und »minus« Buttons, die Karte kann durch den »Standort«-Button auf den aktuellen Standort ausgerichtet werden und durch Berühren und Halten kann die Ansicht verschoben werden.

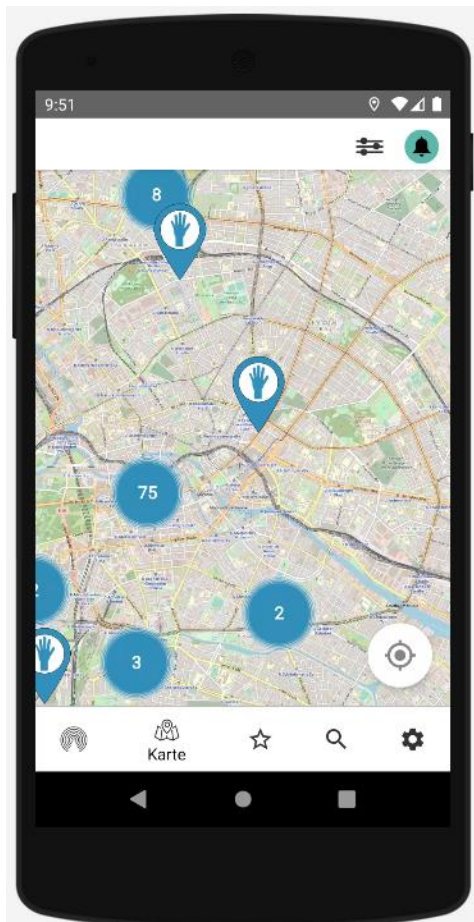
Die Navigationsstruktur beinhaltet neben der Fußleiste am unteren Rand des Bildschirms eine obere Leiste, in der Benachrichtigungen und das Filtermenü erreicht werden können. Das Filtermenü beinhaltet mehrere Filterparameter als ausklappbare Auswahlmöglichkeiten. Beim Öffnen der Filterfunktion erscheint eine Übersicht mit allen fünf Filterparametern. Nutzer:innen sehen auf einen Blick, welche Filterparameter es gibt, sind jedoch nicht mit großer Menge von Informationen konfrontiert. Der gewünschte Parameter kann durch Klicken auf den Pfeil ausgeklappt und eingestellt werden. Eingestellte Filterparameter werden als Kacheln in der oberen Leiste angezeigt, um die Anzeige von Einträgen in der Karten- und Listenansicht transparent und verständlich zu halten. Zudem ist das Setzen von Filterparametern durch eine direkte Texteingabe oberhalb der ausklappbaren Parameter möglich, welches ein schnelles Setzen von Filterparametern ermöglicht.

Insgesamt wurde bei der Präsentation der Informationen auf eine reduzierte Darstellung geachtet, wie es für eine mobile App angemessen ist. Durch den Einsatz von Reitern oder aufklappbaren Inhaltsbereichen, wie beispielsweise im Filtermenü, erhalten die Nutzer:innen dennoch einen Überblick, welche Informationen vorhanden sind. Rein grafische Repräsentationen von Informationen oder Funktionen wurden nur eingesetzt, wenn die Eindeutigkeit der Bedeutung gewährleistet werden kann, wie bspw. die Verwendung eines Wecker-Symbols für den Projektzeitraum oder einem Stern für das Hinzufügen zu den Favoriten.

Die Informationsarchitektur und Navigation wurden auf Grundlage von Nutzeranforderungen erstellt und in Nutzertests eingehend evaluiert und verifiziert. Sowohl die Relevanz und Priorisierung von Informationen wurde untersucht, als auch die Menüführung und -bedienung. Bei grafischen Elementen wurde sichergestellt, dass diese eindeutig zugeordnet werden können. Die genauen Anforderungen aus Interviews, Workshops und Nutzertests können in Abschnitt 3 nachvollzogen werden.

⁵ mit zwei Fingern durch Zusammenziehen oder Auseinanderziehen zoomen

Abbildung 11: Screenshot vom KiezRadar-Prototypen mit Kartenansicht.



2.4.5 Interaktionsdesign

Interaktionsdesign ist eine Komponente des User Experience Designs und eng verbunden mit weiteren Komponenten wie der Informationsarchitektur und dem Navigationsdesign. Interaktionsdesign umfasst die Gestaltung des Dialoges zwischen Produkten/ Systemen und Personen. Dabei liegt der Fokus auf den einzelnen Interaktionsmomenten, wie beispielsweise das Drücken eines Knopfes durch Nutzer:innen und die Rückmeldung, die das System oder Produkt dazu gibt. Ziel ist es, das Interaktionserlebnis zu verbessern.

Der folgende Abschnitt legt dar, welche Aspekte beim Interaktionsdesign im Vordergrund standen.

2.4.5.1 Intuitivität

Ziel der App ist es, schnell und einfach Informationen über relevante Veranstaltungen und Projekte zu finden. Die Listen- und Kartenansicht und die Suchfunktion sind deshalb als zentrale Elemente in der Fußleiste verfügbar. Die App gibt Nutzer:innen Rückmeldung zu eingestellten Filtern, indem sie diese als Kacheln anzeigt. Die Filterfunktion ist ebenso wie die Kacheln im oberen Bereich der App angeordnet. Mit der Platzierung der Filterfunktion und der Kacheln soll unterstrichen werden, dass diese für alle Menü-Masken angewendet wird, bei denen die Kacheln angezeigt werden.

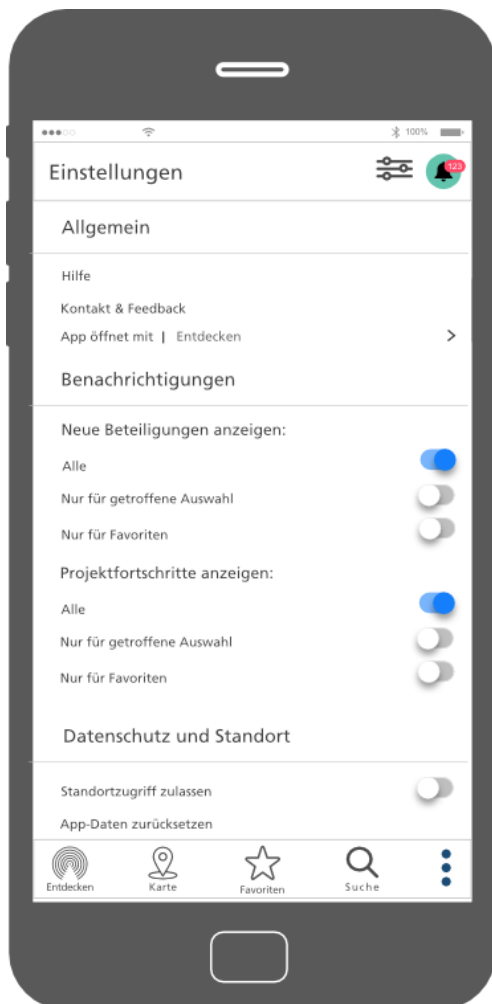
Die Suche bietet den Nutzer:innen eine Möglichkeit, effizient für sie relevante Einträge aufzufinden, ohne sich durch die App navigieren zu müssen.

Für eine hohe Erwartungskonformität bei der Gestaltung von Bedienelementen und Navigationsstrukturen wurde sich an modernen Apps mit hoher Nutzer:innenanzahl sowie an gängigen Heuristiken und Style Guides für Android Apps orientiert. Es wurde zudem darauf geachtet, dass Bedienelemente eindeutig rückmelden, welchen Status sie besitzen, ob sie bspw. aktiv oder inaktiv sind.

2.4.5.2 Benutzerkontrolle

Durch individuelle Einstellungs- und Filtermöglichkeiten und Transparenz in Bezug auf Datenspeicherung soll bei den Nutzer:innen das Gefühl gestärkt werden, dass sie die Kontrolle über die Einstellungen und Datenschutz und Sicherheit haben. Bspw. sollen gesammelte Datenbestände zurücksetzbar sein. Zudem ist eine ausführliche und verständliche Datenschutzhinweise Teil der App. Nutzer:innen wissen dadurch, welche Daten erhoben und genutzt werden.

Abbildung 12: Mockup der Einstellungsseite.



2.4.5.3 Individualisierbarkeit

Die App bietet ein hohes Maß an Individualisierbarkeit und fördert damit eine emotionale Bindung zur App. Inhalte und Benachrichtigungen können auf personenspezifische Bedürfnisse abgestimmt werden und somit besser den persönlichen Interessen der Nutzer:innen gerecht werden.

2.4.5.4 Inspiration

Nutzer:innen benutzen die App, weil sie Veranstaltungen und Projekte in ihrer Nachbarschaft entdecken wollen. Das Interaktionsdesign der App fördert diese Neugierde durch die Entdecken Funktion. Hier sollen Nutzer:innen zum Entdecken von Veranstaltungen und Projekten inspiriert werden und können diese durch die Informationsvorschau in der Listenansicht schnell einordnen.

2.4.5.5 Standortbezogenheit

Die räumliche Verortung der Beteiligungsformate ist ein wesentliches Merkmal der KiezRadar-App. Hier bietet die Nutzung von mobilen Anwendungen Vorteile in Bezug auf die Navigation und Standortinformationen, so dass die Karte als eine eigene Funktion in den Mittelpunkt gerückt wird.

2.4.5.6 Hilfe und Fehlermanagement

Die Hilfe und FAQ sollen die Nutzer:innen bei der Bedienung unterstützen sowie allgemeine Fragen zur KiezRadar-App, dem Projekt und der Pflege beantworten. Die Hilfe ist betont ausführlich und kleinschrittig gehalten, um auch Nutzer:innen abzuholen, die wenig Erfahrung und Übung im Umgang mit der Bedienung von Apps haben. Nutzer:innen können dem Entwicklungsteam außerdem Rückmeldung zu der Anwendung geben, damit diese stetig weiterentwickelt und optimiert werden kann.

3 AP 1: Anforderungserhebung

Die Entwicklung der KiezRadar-App wurde als partizipativer Prozess geplant, bei dem verschiedene Stakeholder (Bürger:innen, Wirtschaft und Verwaltung) aktiv mit einbezogen werden. Die Einbindung der Bürger:innen bildete dabei den zentralen Ausgangspunkt des Anforderungsmanagements. Die erste Einbindung fand in Form eines Bürger:innen-Workshop statt. Auf Basis der Ergebnisse des Workshops wurden Anforderungen in Form von User Stories erstellt und thematisch kategorisiert (bspw. »App Verfügbarkeit«, »Sicherheit und Datenschutz«). Die umfangreiche Sammlung an Anforderungen bildete die Grundlage für die erste Version des Product Backlogs. Um die Umsetzbarkeit der Anforderungen im nächsten Schritt zu bewerten, wurden die Anforderungen identifiziert, bei denen die Berliner Verwaltungsmitarbeiter:innen in einem zweiten Workshop eine Einschätzung geben und ihr Erfahrungswissen einbringen können. Dahinter stecken bspw. die Fragen, ob bestimmte Daten vorhanden sind, welche Teilnahmebedingungen bei den Beteiligungen bestehen und wie detailliert diese dokumentiert werden.

Neben den Workshops wurden Personas für die Ableitung von Anforderungen erstellt, so dass auch unabhängig von den Workshop-Teilnehmer:innen sichergestellt ist, dass die wichtigsten Zielgruppen der App mitberücksichtigt werden.

3.1 Bürger:innen-Workshop

Tabelle 1: Eckdaten des Bürger:innen-Workshops.

Datum	28.11.2019
Zeit	17 – 21 Uhr
Ort	CityLAB Berlin
Teilnehmende	12 Personen

3.1.1 Vorbereitung

Am 28.11.2019 fand der Workshop mit Bürger:innen im CityLAB Berlin statt. Um eine möglichst diverse Teilnehmerschaft in Bezug auf Wohnort, Bildungshintergrund, Alter, technologisches Vorwissen etc. zu erzielen, wurde der Workshop breit beworben. Konkret wurden alle Quartiersmanagements in Berlin und 57 Vereine aus Berlin angeschrieben. Als Quelle für die Suche der Vereine wurde der BerlinFinder genutzt. Bei der Auswahl der Vereine wurde insbesondere auf eine Verwurzelung im Kiez geachtet, bspw. in Form von Nachbarschaftshilfe oder Vergleichbares. Die Vereine und Quartiersmanagements wurden gebeten, den Workshop aktiv in ihrer »Community« zu bewerben.

3.1.2 Konzeption und Durchführung

Bei der Konzeption des Bürgerworkshops wurden zwei primäre Ziele verfolgt – einerseits sollten die Bedürfnisse und Anforderungen der Bürger:innen in Bezug auf die KiezRadar-App möglichst umfangreich erfasst werden und andererseits sollten die Teilnehmer:innen im Sinne eines verkürzten Design-Thinking-Prozesses Ideen für eine konkrete Gestaltung der App entwickeln und in Form eines Low-Fidelity Prototypen umsetzen. Für die systematische Sammlung der Anforderungen sollten die Bürger:innen sich zunächst auf einer abstrakten Ebene mit möglichen Chancen, Herausforderungen, Wünschen und grundlegenden Voraussetzungen für eine erfolgreiche Nutzung der KiezRadar-App

auseinandersetzen. Die Ergebnisse sollten im nächsten Schritt konkretisiert (mögliche Informationsquellen, Anwendungsszenarien) sowie auf den Designaspekt einer App-Lösung gerichtet werden. Auf diese Weise konnten sowohl Anforderungen an die Datenquellen als auch an das Design- und Bedienkonzept der App abgeleitet werden.

Den Ausgangspunkt für diese Überlegungen bildet die zentrale Fragestellung des Workshops:

Wie kann die Information über Beteiligungsmöglichkeiten und lokale politische Ereignisse durch eine App verbessert werden?

Hierfür wurden die Teilnehmer:innen gebeten sich in die Situation hineinzusetzen, in der sie sich zuletzt beteiligt haben, überlegt hatten sich zu beteiligen oder sich im Nachhinein gerne beteiligt hätten. Im nächsten Schritt sollten die Teilnehmer:innen konkrete Aufgaben nach einem vorgebenden Schema erstellen (siehe Beispiel Abbildung 13), die die KiezRadar-App lösen soll, so dass sich Anforderungen in Bezug auf die Funktionen und Zielebene für die App-Gestaltung ableiten lassen. In einem Gruppenprozess wurden die wichtigsten Aufgaben identifiziert.

Abbildung 13: Beispiel zur Definition einer Aufgabe im Bürger:innen-Workshop.



Aufgabe festlegen



5 Min.

Formuliere auf Basis der bisherigen Erkenntnisse eine konkrete Aufgabe, die Du mit Hilfe der KiezRadar-App lösen willst.

Eine Familie mit drei Kindern

Adressat und relevante Fakten

braucht einen Weg, um *Ideen für die Spielplatzneugestaltung einzubringen*

sein/ihr Bedürfnis

weil/obwohl *sie aus Zeitgründen nicht an den klassischen Beteiligungsverfahren teilnehmen können.*

Rahmenbedingung/Herausforderung/Einsicht

Diese wichtigsten drei Aufgaben sind:

- »Verkehrsberuhigende Maßnahmen brauchen einen Weg, um sich über Beteiligungen zu informieren, weil Initiativen gestartet werden sollen oder man sich einer Initiative anschließen möchte.«
- »Eine Familie mit 3 Kindern braucht einen Weg, um Ideen für die Spielzeugneugestaltung einzubringen, weil sie aus Zeitgründen nicht an klassischen Beteiligungsverfahren (vorwiegend am Abend) teilnehmen können.«
- »Eine fahrradbegeisterte Softwareentwicklerin braucht einen Weg, um ihr Interesse an der Entwicklung besserer Fahrradinfrastruktur zu erfüllen, weil sie das Gefühl hat, dass da gar nichts passiert.«

Zu den ausgewählten Aufgaben wurden wiederum vier Lösungsideen entwickelt. Drei dieser Lösungsideen wurden in einem Gruppenprozess weiterentwickelt und in Form eines Low-Fidelity-Prototypen umgesetzt und präsentiert. Dergestalt ließen sich konkrete Lösungs- und Gestaltungsansätze für die drei User Stories ableiten, die aus Sicht der Workshopteilnehmer:innen die größte Relevanz besitzen.

3.1.3 Ergebnisse und Auswertung

Durch den Bürger:innen-Workshop konnten im großen Umfang Anforderungen abgeleitet sowie Ideen und Impulse aufgenommen werden. Die Plattform mein.berlin.de und die Ratsinformationssysteme wurden auch von den Bürger:innen als zentrale Informationsquellen angesehen. Zudem wurde von einzelnen Teilnehmer:innen der Wunsch geäußert, die Angebote des Quartiersmanagements durch die Anbindung der Veranstaltungskalender zu integrieren. Auch die Anbindung von nebenan.de wurde von einzelnen Teilnehmer:innen gefordert.

Eine Idee, die die Standort-Funktion einer mobilen App sehr zielführend einsetzt, ging aus dem Lösungsansatz zu der dritten Aufgabe hervor, die sich mit der Verbesserung der Radinfrastruktur auseinandergesetzt hat. Die Nutzer:innen können sich bei diesem Lösungsansatz am Ende einer Wegstrecke, die sie bspw. mit dem Fahrrad zurücklegen, alle Beteiligungsmöglichkeiten oder sonstigen für sie relevanten Ereignisse an denen sie vorbeigefahren sind in der KiezRadar-App anzeigen lassen. Alternativ wurde die Möglichkeit vorgesehen, dass die Nutzer:innen auch in Echtzeit durch die KiezRadar-App informiert werden, wenn sie an den entsprechenden Ereignissen vorbeifahren. Der Lösungsansatz der Gruppe, die sich mit der Aufgabe der Neugestaltung eines Spielplatzes auseinandergesetzt hat, nutzt ebenfalls die Vorteile einer mobilen Anwendung. Bei diesem Ansatz können die Nutzer:innen die KiezRadar-App einsetzen, um sich direkt am Standort des Spielplatzes mit einer Idee zu beteiligen. Wie einige andere Vorschläge der Teilnehmer:innen geht diese Idee allerdings über den geplanten Fokus und Funktionsumfang der App hinaus, bspw. in dem sie eine direkte Beteiligungsmöglichkeit innerhalb der App vorsieht.

Auf Grundlage der umfangreichen Workshop-Dokumentationen wurden die Anforderungen gesammelt, geclustert, als User Story formuliert und mit Akzeptanzkriterien versehen. Zudem wurden die Anforderungen auf ihre Machbarkeit überprüft. Die gesammelten User Stories können dem Backlog entnommen werden. Zusammen mit den Personas bilden die abgeleiteten User Stories die Basis für die Erstellung der Mockups. Um eine bessere Identifikation und ein tieferes Verständnis der Anforderungen zu fördern, wurden diese soweit möglich den verschiedenen Zielgruppen der erstellten Personas (siehe Abschnitt 3.4) zugeordnet.

3.2 Workshop mit Verwaltungsmitarbeiter:innen

Tabelle 2: Eckdaten des Workshops mit Verwaltungsmitarbeiter:innen.

Datum	10.09.2020
Zeit	14 – 17 Uhr
Ort	digital
Anmeldungen	11 Personen

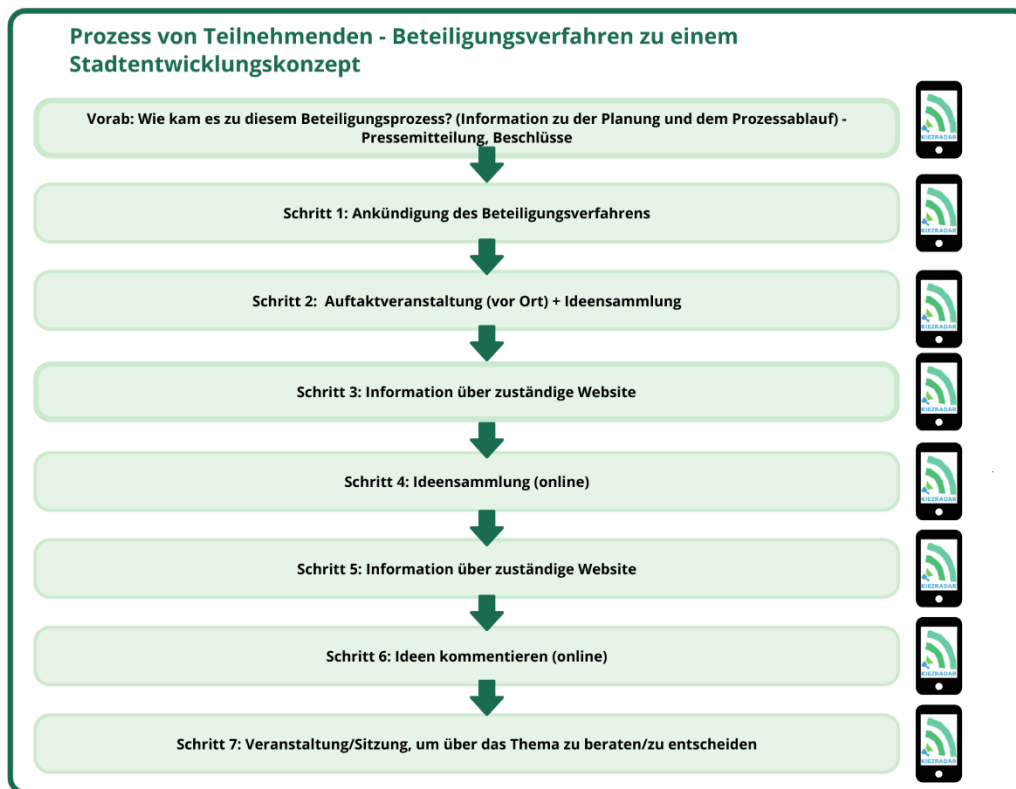
3.2.1 Vorbereitung

Im Fokus der Bewerbungen des Workshops standen Verwaltungsmitarbeitende, deren Arbeit im Bezug zu Beteiligungsverfahren und Bezirkspolitik in Berlin steht. Unter anderem wurden die BVV-Büros der Bezirke, die Bezirksämter, die Senatsverwaltungen für Stadtentwicklung, -für Umwelt, Verkehr und Klimaschutz, -für Inneres und Sport, -für Gesundheit, Pflege und Gleichstellung sowie die Stadtentwicklungsämter der Bezirke informiert. Der Workshop war ursprünglich für den 12.03.2020 als Präsenzworkshop im CityLAB Berlin geplant und musste aufgrund der Corona-Pandemie kurzfristig abgesagt werden. Am 10.09.2020 wurde der Verwaltungsworkshop digital nachgeholt. Um bei dem digitalen Workshop-Format die Belastung der Teilnehmer:innen gering zu halten, wurde der Inhalt gegenüber der ursprünglichen Planung um eine halbe Stunde gekürzt. Aus diesem Grund wurde den Teilnehmer:innen vorab ein Bogen mit Fragen zu ihren bisherigen Erfahrungen mit Bürgerbeteiligungsprozessen geschickt. Dieser enthielt u. a. Fragen zu den dabei genutzten digitalen Plattformen, den beteiligten Personen/ Rollen, Optimierungspotentialen und Aufwandstreibern.

3.2.2 Konzeption und Durchführung

In dem Workshop sollten die Verwaltungsmitarbeiter:innen die Möglichkeit bekommen, ihre Gedanken, Wünsche, Ziele und Bedenken zu der späteren KiezRadar-App zu diskutieren und Ideen zu möglichen Anwendungsszenarien oder Eigenschaften der App entwickeln zu können. Zu diesem Anlass wurde zunächst der aktuelle Prozess zur Information einer Beteiligung auf der Plattform mein.berlin.de sowie zur Information über eine BVV Ausschusssitzung über das Ratsinformationssystem des Bezirks Friedrichshain-Kreuzberg vorgestellt. Dem wurde gegenübergestellt, wie Beispielprozesse mittels der KiezRadar-App umgesetzt werden sollen. Die Einbindung der KiezRadar-App wurde im Weiteren anhand des ersten Beispiels, der Beteiligung über mein.berlin.de, vertieft. Hierbei wurden für jeden einzelnen Schritt die Möglichkeiten und Implikationen der Prozessintegration der KiezRadar-App diskutiert. Die Teilnehmenden sahen bei allen Teilschritten des Prozesses eine sinnvolle Integrationsmöglichkeit der KiezRadar-App (siehe Abbildung 14)

Abbildung 14: Ergebnisse zur Prozessintegration aus dem Verwaltungsworkshop.



Der zweite Schwerpunkt des Verwaltungsworkshops lag darauf, die Umsetzbarkeit der Ergebnisse und Ansätze aus dem Bürgerworkshop zu diskutieren. Dafür wurden im Vorfeld die Anforderungen aus dem Bürgerworkshop identifiziert, bei denen Unklarheiten in Bezug auf die Machbarkeit bestanden bzw. bei denen die domainspezifischen Kenntnisse der Verwaltungsmitarbeiter:innen berücksichtigt werden sollten. Für eine bessere Abstraktion der Anforderungen wurde auch hier auf die erstellten Personas zurückgegriffen. Je Persona wurden sieben bis acht User Stories vorgestellt, die seitens der Teilnehmer:innen nach folgenden Kategorien klassifiziert werden sollten: sofort umsetzbar, abhängig von... und schwer umzusetzen. Abbildung 15 bis Abbildung 17 zeigen die Anforderungen der jeweiligen Personas.

Abbildung 15: Ausgewählte Anforderungen der Persona Nora.

Anforderungen



Abbildung 16: Ausgewählte Anforderungen der Persona Florian.

Anforderungen

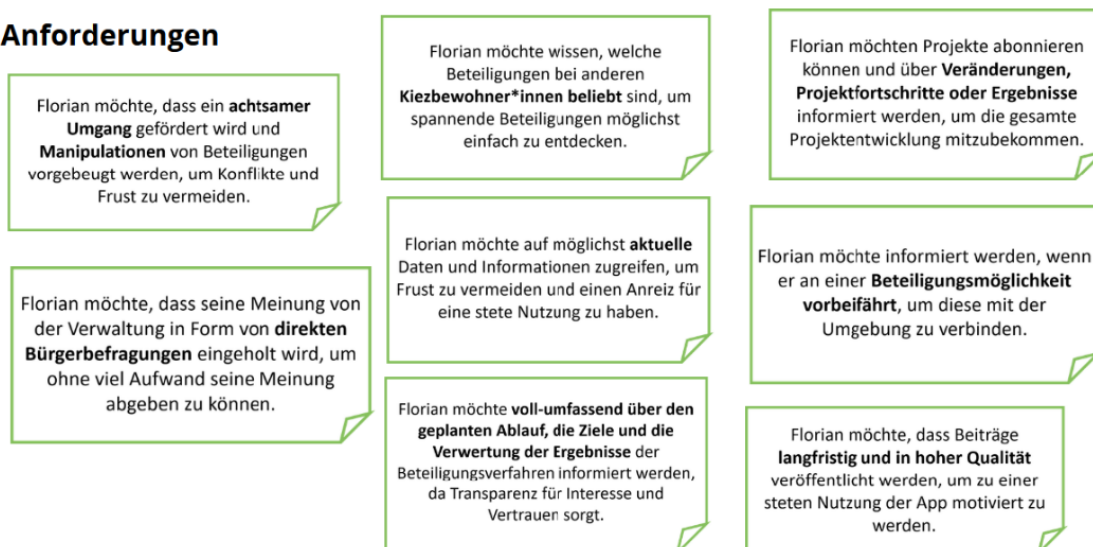
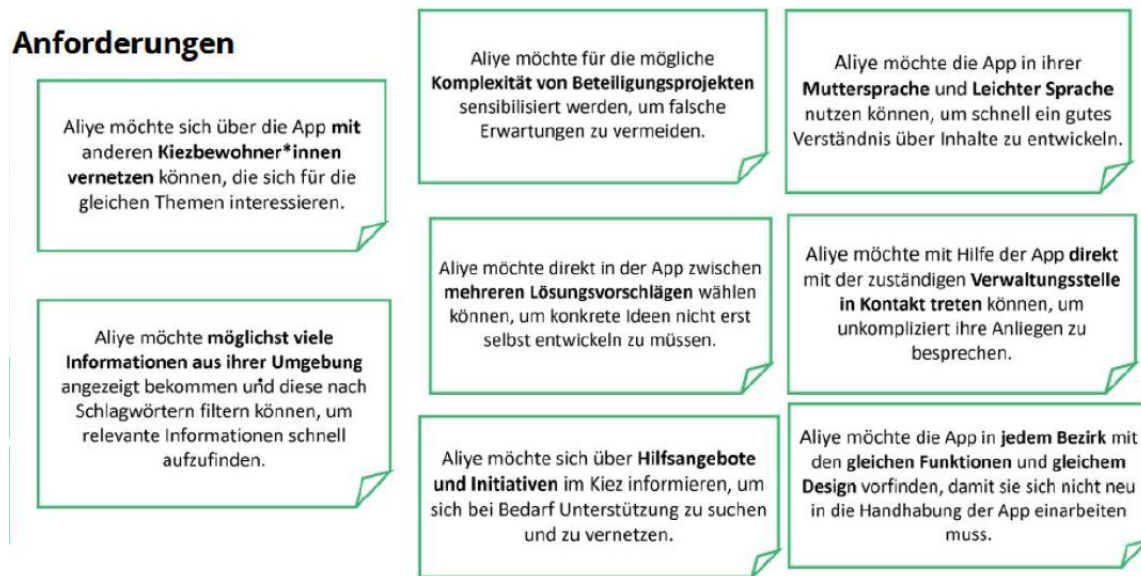


Abbildung 17: Ausgewählte Anforderungen der Persona Aliye.



Aus dem Workshop ergab sich die folgende Verteilung, wobei eine große Zahl der Anforderungen aufgrund fehlender Zeit nicht eingeordnet werden konnte.

Tabelle 3: Ergebnis der Bewertung der Umsetzbarkeit von Anforderungen durch Verwaltungsmitarbeiter:innen.

	User Story	Dahinterliegende Fragestellung	Hinweis der Teilnehmer:innen
Sofort umsetzbar	Nora möchte anonym an Beteiligungen teilnehmen, um keine Daten weitergeben zu müssen.	Gibt es Beteiligungsformate die anonym nutzbar sind? Müssen dafür Voraussetzungen erfüllt werden? Wie hoch ist der Anteil?	"BVV-Sitzungen nicht anonym"
	Nora wünscht sich eine Offenlegung der bei der KiezRadar-App genutzten Daten, um sie auch für andere Projekt nutzen zu können.	Ist die Offenlegung der in der App wiedergegebenen Daten oder von Projekt-Daten geplant? Wie und in welchem Umfang erfolgt dies bereits? (Stichwort Open Data).	"Aufbereitung in einem Ergebnisdokument"
Abhängig von...	Nora möchte, dass die App eine Vielzahl an Medien und Inhalten von anderen Plattformen einbindet, damit sie eine möglichst breite Informationsbasis zu Beteiligungsangeboten erhält.	Welche Plattformen sind relevant? Wie lassen sie sich abbilden? Wo finden sich überall relevante Infos? Welche Plattformen sollen nicht berücksichtigt werden?	"Personelle Hürden (bspw. redaktionelle Arbeit)." "Datenqualität und Einheitlichkeit"
	Nora möchte eigene Initiativen und Beteiligungsformate gründen und anzeigen lassen, um die Bürger:innen zu vernetzen und ihre Interessen abzubilden.	Ist das als mögliche Funktion von der Verwaltung gewünscht? Sinnvoll?	"Kontrollschwierigkeiten" "Aufgabe der Anlaufstellen"
	Florian möchte, dass ein achtsamer Umgang gefördert wird und Manipulationen von Beteiligungen	Gibt es sowas wie ein Code of conduct? Gibt Strategien gegen	"Problem der anonymen Beteiligung"

	User Story	Dahinterliegende Fragestellung	Hinweis der Teilnehmer:innen
	vorgebeugt wird, um Konflikte und Frust zu vermeiden.	Manipulationen von Beteiligungen oder Trolling?	"manuelle/händische redaktionelle Betreuung der Verhaltensregeln (unpassende Kommentare werden gelöscht)"
	Florian möchte, dass seine Meinung von der Verwaltung in Form von direkten Bürgerbefragungen eingeholt wird, um ohne viel Aufwand seine Meinung abgeben zu können.	Welche Beteiligungsformate gibt es? Gibt es direkte Formate mit ja/nein oder Vorauswahl durch die Verwaltung? Wann ja wie häufig?	"hoher Aufwand (Kosten, Planung, Vorlauf)" "Wie ist die Repräsentativität der Umfrage, wenn Menschen sich z. B. nicht beteiligen wollen/ interessiert sind"
Schwer umsetzbar	Florian möchten Projekte abonnieren können und über Veränderungen, Projektfortschritte oder Ergebnisse informiert werden, um die gesamte Projektentwicklung mitzubekommen.	Werden detaillierte Prozessfortschritte und Ergebnisse veröffentlicht, um Entwicklungen abzubilden?	"Zuordnung von Dokument (Beschluss) zu Projekt – schwierig!" "Unklarheit über zukünftige Entwicklungen (möglicherweise)" "lange Zeiträume der Prozesse sind evtl. ein Problem der Darstellung"

3.2.3 Ergebnisse und Fazit

Die Einschätzungen zur Machbarkeit der User Stories durch die Verwaltungsmitarbeiter:innen wurde anschließend in das Backlog eingearbeitet. Eine wichtige Erkenntnis, die durch den Input der Verwaltungsmitarbeiter:innen gewonnen werden konnte ist, dass diese ein besonders hohes Potential für die KiezRadar-App sehen, Informationen entlang des Lebenszyklus eines Projekts miteinander zu verknüpfen. Gemeint sind hierbei die Projektinformationen, die vor dem Beginn der eigentlichen Beteiligungen entstehen, also bspw. zurückliegende Entscheidungen und Informationen aus der BVV. Diese Informationen können z. B. als zurückliegende Meldungen zum Projekt in der KiezRadar-App angezeigt werden. Die Voraussetzungen für eine solche Verknüpfung der Informationen erfüllen die derzeitigen Datenquellen allerdings nicht.

Neben den aufgenommenen Anforderungen aus Sicht der Verwaltungsmitarbeiter:innen konnten die Erkenntnisse zu der Organisation von Beteiligungen in Berlin vertieft und offene Fragen, bspw. in Bezug auf die Inhalte der Beteiligungskonzepte, geklärt werden.

Insgesamt fiel das Fazit zu dem digitalen Format des Verwaltungsworkshops positiv aus. Workshopinhalte, die eine kreative und interaktive Mitarbeit der Teilnehmenden vorausgesetzt haben, waren dennoch eine besondere Herausforderung. Hier mussten teilweise längere Zeiten abgewartet werden, um bspw. eine Reaktion der Teilnehmenden auf Fragen zu erhalten. Eine höhere und schnellere Beteiligung wurde erzielt, wenn die Teilnehmenden Input nicht verbal, sondern durch das Ausfüllen von Post-Its, das Verschieben von

Inhalten, das Setzen von Kreuzen etc. abgeben konnten. Ein Präsenzworkshop ist dem digitalen Format, wenn möglich aber vorzuziehen.

Von den vorab zugesendeten Projektsteckbriefen wurden vier ausgefüllt zurückgesendet. Zu den im Beteiligungsprozess genutzten digitalen Plattformen zählten mein.berlin.de und Adhocracy Plus. In einem Beteiligungsprojekt zu einer Radverkehrsanlage in Berlin wurde keine digitale Plattform genutzt. Kritisiert wurden an den Beteiligungsprozessen eine fehlende Rückmeldung zum Stand der Umsetzung sowie adäquate Prozesse, um derart komplexe Probleme aufzudröseln und die Ergebnisse wiederum zielführend zu einer Lösung zusammenzuführen. Verbesserungspotentiale im Beteiligungsprozess wurden in Form einer besseren Absprache mit den Betroffenen vor Ort, einer umfangreichen Kommunikation des Inputs durch die Bürger:innen, größerer Entscheidungsräume und einer Ausweitung des Online-Angebots und der Online-Werbung ausgemacht.

3.3 Einbindung von Akteuren aus der Wirtschaft

Anders als zunächst im Projektantrag geplant wurde in Absprache mit der Senatskanzlei davon abgerückt, die Wirtschaft in Form eines Workshops aktiv einzubinden. Die Gründe hierfür waren unter anderem, dass die Einbindung der Plattform mein.berlin.de und der Ratsinformationssysteme der Bezirke bereits zu Projektbeginn feststand. Die übrigen Ressourcen sollten dabei, den übergeordneten Projektzielen von KiezRadar entsprechend, auf die Berücksichtigung von bürgernahen Daten verwendet werden. Im Projektverlauf wurde demonstriert, dass die Einbindung von heterogenen Daten aus unterschiedlichen Quellen und mit verschiedenen Datenformaten möglich ist, so dass auch Daten aus Wirtschaftskontexten grundsätzlich integriert werden können. Mögliche Anwendungsszenarien hierfür wären u.a. lokale Anbieter mit Angeboten, bspw. der Bäckerladen um die Ecke, der Brötchen und Kuchen zum Feierabend zu einem geringen Preis anbietet. Weiter wäre auch die Einbindung privater Social-Media-Plattformen ein denkbare Szenario. Ein Austausch mit den Betreibern der Social-Media-Plattform »nebenan.de« wurde zwar angestrebt, kam aber aufgrund der Corona-Pandemie nicht zustande.

3.4 Personas

Personas helfen, die Entwicklungen einer App nicht nur aus der Perspektive einer Projektleiterin oder eines technischen Entwicklers zu betrachten, sondern die Sicht verschiedener Interessens- und Nutzergruppen einzunehmen. Dadurch lassen sich wichtige Anforderungen und Bedürfnisse ableiten, wie bspw. Mehrsprachigkeit oder barrierefreies Design. Die Erstellung von Personas fördert daher eine nutzerzentrierte Entwicklung, auch weil sie hilft, die Motive, Erfahrungen, Verhaltensweisen und Ziele der Nutzer:innen besser zu verstehen. Ziel der Persona-Erstellung ist es also, dass die KiezRadar-App für die verschiedenen Zielgruppen möglichst gut und ohne Einschränkungen nutzbar ist.

Die ausgewählten Personas sollen die drei Nutzergruppen repräsentieren, die wir als Projekt-Team für die KiezRadar-App als besonders relevant eingestuft haben:

- Technik-affine Nutzer:innen
- Politisch vernetzte Berliner:innen
- Alteingessene Kiezbewohner:innen

3.4.1 Methodik

Personas sind fiktive Charaktere, die eine Nutzergruppe eines Produktes repräsentieren. Personas helfen dabei, die Bedürfnisse, Erfahrungen, Verhaltensweisen und Ziele der Nutzer:innen zu verstehen. Typischerweise basiert eine Persona auf realen Daten, die von mehreren Personen gesammelt und in einen fiktiven Charakter übersetzt werden. Die Erstellung von Personas ermöglicht es, potenzielle Nutzer:innenprofile zu verstehen und fördert ein nutzerzentriertes Design.

Personas sind eine reichhaltige Darstellung von Nutzer:innen, die über reine demografische Details hinausgehen und Einblicke in Einstellungen und Verhaltensweisen bieten. Personas dienen außerdem dazu, Nutzergruppen für das Designteam einprägsamer zu machen und Empathie aufzubauen.

Personas enthalten typischerweise Informationen über

- Demografische Angaben, z. B. Alter, Einkommen, Beruf
- Persönliche Details, z. B. Kurzbiografie, Name, Bild
- Einstellungsbezogene und/oder kognitive Details, z. B. Einstellung gegenüber neuen Anwendungen
- Ziele und Motivation für die Nutzung des Produkts
- Verhaltenstechnische Details in Situationen mit dem Produkt

3.4.2 Umsetzung

Die Nutzer:innen-Personas entstanden im Nachgang an den Bürger:innen-Workshop im November 2019. Der Workshop, zusammen mit Interviews, lieferte die Grundlage der Personas. Es wurden fünf Personen im Alter von 25 – 45 Jahren, die potentielle Nutzer:innen darstellen, gemäß einem vorher erstellten Interviewleitfaden interviewt. Die befragten Personen wohnen größtenteils in Berlin.

Ein ähnliches Verfahren war für die Erstellung einer Verwaltungspersona vorgesehen. Allerdings konnte durch einen Aufruf nur eine Person für ein Interview gewonnen werden, die mithilfe eines vorgefertigten Interviewleitfadens befragt wurde. Da ein Interview nicht ausreichend für die Erstellung einer Persona ist, fehlt die Verwaltungspersona. Im Workshop mit Verwaltungsmitarbeiter:innen wurden jedoch Anforderungen aus der Verwaltung identifiziert und im Entwicklungsprozess berücksichtigt.

Die Anforderungen aus dem Bürger:innen-Workshop wurden den verschiedenen Personas zugeordnet und durch die Anforderungen ergänzt, die aus den Interviews mit den Repräsentant:innen der Nutzergruppen abgeleitet werden konnten. Diese Anforderungen wurden in Form von User Stories ebenfalls in das Product Backlog übernommen. Ferner wurde das Product Backlog um eigene technische und fachliche Anforderungen ergänzt.

3.4.3 Ergebnisse

Im Folgenden werden die erstellten Personas im Detail vorgestellt. Sie folgen dabei alle demselben Schema und sind unterteilt in: Allgemeine Vorstellung, Charakteristika, Anforderungen und Bedürfnisse, Ziele und Aufgaben und Motivation.

Tabelle 4: Detailübersicht der Persona »Technik-affine Nutzerin«.

Technik-affine Nutzerin



Nora
 33 Jahre
 Kommt aus Rostock, lebt im Prenzlauer Berg
 ledig
 Netzwerkadministration und IT—Support

Motto:

»Eine gute Anwendung ist Open Source und achtet auf Datenschutz.«

»Ich möchte meine Fähigkeiten einsetzen, um eine gerechtere Gesellschaft mitzugestalten.«

Charakteristika

Nora lebt nun schon seit 10 Jahren in Berlin und arbeitet hier als Netzwerkadministratorin und IT-Supporterin. Sie ist ein energischer und umtriebiger Mensch, vernetzt sich gut und hat eine offene und neugierige Lebenseinstellung. Es ist ihr wichtig, eigenständig und unabhängig von großen Konzernen zu leben, wofür sie technische Hürden gerne auf sich nimmt, wie beispielsweise die Benutzung eines eigenen Servers. Gegenseitige Unterstützung und Teilen von Fähigkeiten, Wissen und Infrastruktur haben eine hohe Priorität für sie. Sie nimmt deshalb regelmäßig an regionalen und überregionalen Veranstaltungen und Aktionen der Netzpolitik-Community teil und setzt sich für Open Data und Open Source ein. Immaterielle Werte, wie das Anstreben eines guten Lebens, stehen für Nora über materiellen. Durch Ausbildung und Beruf besitzt sie fortgeschrittene technische Fähigkeiten, beherrscht mehrere Programmiersprachen (u. A. C++, Java) und erkundet neugierig technische Hintergründe, um ihr Verständnis zu erweitern. In ihrem Freundeskreis gibt sie Impulse und ist eine kritische Stimme, die über technische Zusammenhänge, wie Sicherheitslücken, aufklärt und gerne auch mal die Rolle der Kommunikatorin übernimmt.

Anforderungen und Bedürfnisse

- Informationskanäle: rein digital, Videos, Podcasts, Blogs, Newsfeeds, E-Books (Sachbücher)
- Informationsbedürfnis: Hoch, informiert sich über aktuelle Nachrichten, Info-Junkie: hohes Informationsbedürfnis wird durch die Kombination vieler digitaler Medien gestillt.
- Sicherheitsbedürfnis: In Bezug auf Datensicherheit hoch, sonst eher gering, offen für Neues
- App-Nutzung: Funktion ist wichtiger als Design, sehr überlegte Installation von Apps, aber nutzt relativ viele Apps, temporäre Installation von Apps, probiert auch mal was aus, bezahlt auch gerne für eine App. Funktion vor Design: bei der App-Nutzung ist die Funktionalität entscheidend.

Ziele und Aufgaben

- Interessen/Hobbies: DIY-Projekte (Löten, Arduino, Microcontroller, 3D-Druck u. Ä.) elektronische Musik, Organisation und -teilnahme an Open Air Musik Festivals
- Lebensziele: gute Beziehungen pflegen, an Kultur und Stadtleben teilhaben, gesellschaftlich

Motivation

Vorbilder: Edward Snowden, Margaret Hamilton, Pipi Langstrumpf, Linus Neumann, Mai Thi Nguyen-Kim

- oder politisch wirken für ein gerechteres Zusammenleben
- Politische Engagement: Demos, Clubvereinigung (Bündnis gegen das Sterben der Berliner Clubkultur), aktiv im Verein für feministische Netzpolitik

Tabelle 5: Detailübersicht der Persona „Politisch vernetzter Berliner“.

Politisch vernetzter Berliner



Florian
25 Jahre
Kommt aus Freiburg, lebt in Neukölln
ledig
Student (Kulturwissenschaften)

Motto:

- »Begegnung und Austausch sind mir wichtig.«
- »Ich wünsche mir transparente Entscheidungsfindungen und eine lebendige Gemeinschaft.«

Charakteristika

Florian studiert Kulturwissenschaft und ist für sein Studium nach Berlin gezogen. Er nutzt seine Freizeit gerne, um sich gesellschaftlich einzubringen und engagiert sich politisch in verschiedenen Initiativen. Dort wird er für seine reflektierte und unterstützende Persönlichkeit geschätzt. Er hat viele Interessen, ein großes Verantwortungsgefühl und eine Leidenschaft dafür, Dinge anzupacken, ist jedoch ein wenig chaotisch. Ein gutes Zusammenleben hat einen hohen Stellenwert für Florian, weshalb ihm transparente Entscheidungsfindungen, rationale Debatten, Orientierung am Gemeinwohl und Toleranz sehr wichtig sind. Er lebt gerne in Berlin, da die Stadt viele Möglichkeiten und Facetten bietet, findet aber, dass viele Dinge verbessert werden können. Er wünscht sich zudem, dass sich mehr Menschen an der Gestaltung des urbanen Zusammenlebens beteiligen. Auf seinem Smartphone nutzt er verschiedene Apps, jedoch immer mit einem kritischen Blick auf Datenschutz, wofür er sich auch mal in technische Details einarbeitet.

Anforderungen und Bedürfnisse

- Informationskanäle: digital und analog, Videos und Audio (Podcasts), Bücher (auch Sachbücher), Zeitung
- Informationsbedürfnis: hoch, aber nimmt nicht immer teil, vor allem im Kontext »Wem gehört die Stadt/Stadt-Architektur«
- Sicherheitsbedürfnis: nicht ängstlich, lebt in einem sicheren Land, offen für Neuerungen, Freiheiten wichtiger als Sicherheiten
- App-Nutzung: eher ausgewählt Nutzung, die regelmäßig genutzt werden, und die gut funktionieren sollen, hat keine Lust sich groß einzuarbeiten.

Ziele und Aufgaben

- Interessen/Hobbies: Klettern, Lebensmittel retten, Holzbau, Imkern

Motivation

Vorbilder: Hannah Arendt, Noam Chomsky, Margarete Stokowski

- Lebensziele: Was gutes Hinterlassen für die Gemeinschaft, gemeinschaftliches Leben, Familie
- Politische Engagement: ehemals bei den Grünen, Mitarbeit bei einem Berliner Volkstscheid zum Thema Radinfrastruktur, geht regelmäßig auf Demos, Vorstand eines Vereins zur Berliner Stadtraumkultur

Tabelle 6: Detailübersicht der Persona »Alteingesessene Kiezbewohnerin«.

Alteingesessene Kiezbewohnerin



*Aliye
49 Jahre
Kommt aus Erzurum (Türkei), lebt im Wedding
geschieden, 3 Kinder
Laborhelferin*

Motto:

»Meine direkte Umgebung möchte ich mitgestalten.«

Charakteristika

Aliye ist Laborhelferin und ein lebhafter, herzlicher Mensch mit einer pragmatischen Herangehensweise. Sie ist gut darin zu vernetzen und setzt auf persönliche Kontakte und ihre soziale Ader, um Lösungen zu finden. Obwohl ihr Deutsch nicht perfekt ist findet sie sich gut im Alltag zurecht. Formulare lässt sie jedoch von ihren Kindern ausfüllen, da sie einige Wörter nicht versteht. Aliye wohnt seit Jahrzehnten in einem Weddingener Viertel und trägt dort zu einem lebhaften Kiez bei. Ihre Familie ist ihr sehr wichtig, genauso wie beruflicher Erfolg und gute Bildung für ihrer Kinder. Vernetzung und Engagement in der Nachbarschaft machen einen großen Teil ihrer Lebensqualität aus. Aliye kann gut mit Smartphone und Tablet umgehen, ihre Fähigkeiten am PC sind dagegen begrenzter. Ihre Freund:innen schätzen sie als geduldige Zuhörer:in, für ihre guten Ratschläge und ihre Verlässlichkeit und sehen sie als Vorbild.

Anforderungen und Bedürfnisse

- Informationskanäle: Radio, Türkische Online-Zeitung, Austausch mit Bekannten
- Informationsbedürfnis: gering, möchte über Dinge informiert werden, die sie und ihr Umfeld betreffen
- Sicherheitsbedürfnis: gering in Bezug auf Datensicherheit, wünscht sich Sicherheit in Bezug auf das Kiezleben, das durch Gentrifizierung bedroht wird
- App-Nutzung: Besitzt ein Smartphone, um über Instant-Messenger zu kommunizieren. Benutzt nur Apps mit einfacher und intuitiver Benutzeroberfläche, da sie sich sonst schnell überfordert fühlt.

Ziele und Aufgaben

- Interessen/Hobbies: Backen, Kochen, mit Freund:innen treffen, töpfern, reisen, shoppen
- Lebensziele: ein gutes Leben für sich, Familie und Freunde,

Motivation

Vorbilder: Mustafa Kemal Atatürk, Orhan Gencebay, Halide Edip Adivar, Meryl Streep

- Politische Engagement: ehemals Vorstand CHP, setzt sich in einer Nachbarschaftsinitiative gegen Verdrängung aus ihrem Kiez ein und nimmt an Aktivitäten der türkischen Gemeinde in ihrer Nachbarschaft teil

Auf die Anforderungserhebung und Erstellung von Personas und User Stories folgte die Bewertung der Anforderungen in Bezug auf deren Umsetzbarkeit durch Verwaltungsmitarbeiter:innen. Diese konnten durch ihre Praxiserfahrung identifizieren, welche Anforderungen durch beispielsweise die eingeschränkte Verfügbarkeit von Daten oder Teilnahmebedingungen problematisch sind. Die Bewertung der Umsetzbarkeit der Anforderungen wurde ins Product Backlog aufgenommen.

4 AP 1: Backlog und Anwendungsszenarien

Die Anforderungen aus den zwei Workshops mit Bürger:innen und Verwaltungsmitarbeiter:innen, aus dem Prozess der Persona-Erstellung und aus den Nutzertests wurden in das Product Backlog aufgenommen. Zudem wurde es um die Bewertung der Umsetzbarkeit der Anforderungen ergänzt. Ferner wurden dem Product Backlog technische und fachliche Anforderungen hinzugefügt, die sich bspw. aus gesetzlichen Vorgaben ableiten.

Bei der Definition und Gliederung der Anforderungen wird ein nutzerzentrierter Ansatz verfolgt, bei dem die Gestaltung des gesamten Prozessablaufs aus Sicht der späteren Anwender:innen bzw. eingebundenen Stakeholder strukturiert wird. Entlang einer User Journey (Nutzungsreise) werden die wichtigen Phasen, die in der Interaktion mit den Nutzer:innen auftreten können dargestellt: die erforderliche Nutzeraktion, die Handlungsalternativen und die konkreten Berührungspunkte mit der App bspw. in Form von Menüpunkten.

Des Weiteren werden Epics formuliert, die Anforderungen auf einer höheren Abstraktionsebene beschreiben. Die Epics umfassen die einzelnen Kontakt-Ereignisse sowie übergreifende Funktionalitäten im gesamten Prozess, die für die Nutzung und Akzeptanz der KiezRadar-App erforderlich sind. Anhand der User Journey und den Epics lassen sich systematisch weitere Anforderungen ableiten. Für ein nutzerzentriertes Anforderungsmanagement werden die Anforderungen als User Stories formuliert. Dadurch, dass eine User Story die drei W-Fragen **wer**, **was** und **warum** beantwortet, wird die Priorisierung der Anforderungen erleichtert und nutzergruppenorientierte Entwicklung gefördert.

4.1 User Journey

Tabelle 7: User Journey.

Phase	Nutzeraktionen	Optionen	Touchpoints
Kenntnis	Erstmalig über die App informieren oder informiert werden.	Information via: <ul style="list-style-type: none"> ▪ Freunde ▪ Webseite und Social Media (FOKUS, CityLAB Berlin, Seiten der Stadt Berlin wie mein.berlin.de) ▪ CityLAB Berlin ▪ Werbung ▪ Zeitungen, Magazine etc. ▪ Per E-Mail (ehemalige Teilnehmer:innen) 	
Installation	App auffinden, herunterladen und installieren.	Download via: <ul style="list-style-type: none"> ▪ Google Play; ▪ direkt aus dem Repository (Installation muss dann aber vom Nutzer manuell ausgeführt werden); ▪ F-Droid-Store 	
Anwendung	Anpassung der App an die eigenen Bedürfnisse.	Es können diverse Einstellungen vorgenommen werden, u.a.: <ul style="list-style-type: none"> ▪ Allgemein <ul style="list-style-type: none"> – Hilfe 	<ul style="list-style-type: none"> ▪ Einstellungen ▪ Filter

Phase	Nutzeraktionen	Optionen	Touchpoints
		<ul style="list-style-type: none"> – Kontakt und Feedback – App öffnet mit... ▪ Startseite festlegen ▪ Benachrichtigungen festlegen (am besten Checkboxes) ▪ Über alle neuen Beteiligungen und Fortschritte informieren <ul style="list-style-type: none"> – Nur bei den Favoriten – Nur Projektfortschritte – Nur neue Beteiligungen, die dem Filtern entsprechen <p>Es können diverse Filter gesetzt werden, u. a.:</p> <ul style="list-style-type: none"> ▪ Entfernung ▪ Zeitraum ▪ Kategorien ▪ Bezirke ▪ Beteiligung ▪ Projektstatus 	
	Relevante Inhalte (Informationen und Beteiligungen) entdecken.	Mittels der Suche, der Kartenfunktion oder innerhalb des Menüpunkts »Entdecken« können Informationen gefunden werden zu: <ul style="list-style-type: none"> ▪ aktuellen Beteiligungsformaten ▪ Veranstaltungen und Beratungs- und Freizeitangeboten im Kiez ▪ Öffentliche Sitzungen der BVV ▪ evtl. kleine Anfragen 	<ul style="list-style-type: none"> ▪ Suche ▪ Kartenfunktion ▪ Entdecken
	Das Erhalten neuer Informationen erwirken.	Durch das Hinzufügen von Beteiligungen oder anderen Formaten zu »Favoriten« können Projektfortschritte verfolgt werden. Durch das Einstellen/ Zustimmen Benachrichtigungen erhalten zu wollen, werden die Nutzer:innen über Projektfortschritte, neue Beteiligungen, Termine oder Angebote informiert.	<ul style="list-style-type: none"> ▪ Favoriten ▪ Benachrichtigungen
	Teilnehmen an einem oder mehreren Beteiligungsformaten.	Die Nutzer:innen klicken auf Teilnehmen und füllen die notwendigen Informationen aus oder werden auf die spezifische Projektseite weitergeleitet. (Evtl. optional, sie fügen die Veranstaltung dem Kalender hinzu).	<ul style="list-style-type: none"> ▪ Teilnehmen-Button ▪ Weiterleitung auf spezifische Seite
Deinstallation	Deinstallation der App.	Variiert je Betriebssystem.	

4.1.1 Prozessschritte der Nutzung

Tabelle 8: Übersicht der Epics der Prozessschritte der Nutzung.

#	Epic	Beschreibung
1	Verfügbarkeit/Installation	Sämtliche Prozesse, die für die Installation der App notwendig sind.

#	Epic	Beschreibung
2	Informationsmenge steuern	Funktionen der App zur Filterung der Einträge, Speicherung der Filter und Aufruf von Detailinformationen.
3	Informationen zu Beteiligungsformaten und Veranstaltungen	Funktionen zum Auffinden von Informationen (Suche, Liste, Karte) zu Beteiligungsformaten und Veranstaltungen.
4	Kartenfunktion und standortbezogene Nutzung	Funktionen zur standortbasierten Nutzung der KiezRadar-App. Kartenbasierte Anzeige von Informationen.
5	Beteiligungen wahrnehmen	Informationen oder Weiterleitung auf die entsprechenden Plattformen, die eine Wahrnehmung von Beteiligungen und Veranstaltungen ermöglichen.
6	Projektergebnisse verfolgen	Alle Prozesse und Funktionen, die das Verfolgen der Projektergebnisse ermöglichen (z. B. zu Favoriten hinzufügen, Benachrichtigungen über neue Fortschritte und Ereignisse).
7	Interoperabilität/Integration	Interaktion der KiezRadar-App mit anderen Plattformen und Diensten.
8	Kommunikation	Funktionen und Informationen zu Kommunikationsmöglichkeiten.

4.1.2 Supportfunktionen

Tabelle 9: Übersicht der Epics der Supportfunktionen.

#	Epic	Beschreibung
9	Hilfe und Datenschutz	Hilfestellungen zur Nutzung der App, zum Hintergrund und FAQ, Support-Prozesse zu technischen Supportprozessen, Informationen zum Impressum der App sowie Informationen und Einstellungsmöglichkeiten zum Datenschutz.
10	Einstellungsmöglichkeiten	Alle individuellen Einstellungsmöglichkeit zur Nutzung der App.
11	Barrierefreiheit	Apps von Trägern öffentlicher Gewalt müssen dem Behindertengleichstellungsgesetz (BGG) nach barrierefrei sein (§ 12). Apps sollen von allen Menschen mit Behinderungen bedient werden können.
12	Navigation und Bedienung (Usability)	Unterstützung einer intuitiven und aufgabenangemessene Navigation sowie erwartungskonforme Gestaltung von Elementen.

4.2 Prozessschritte der Nutzung

4.2.1 Verfügbarkeit und Installation

Tabelle 10: Umgesetzte User Stories aus dem Epic »Verfügbarkeit und Installation«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E01.01	Die datenkritischen Nutzer:innen möchten die KiezRadar-App auch ohne die Nutzung von Google Play installieren können.		Der Prototyp ist manuell installierbar: https://kiezradar.fokus.fraunhofer.de/blog/2021-02-17-apk/
E01.02	Kritische Nutzer:innen möchten die App als Open Source-Software transparent einsehen können, da sie Wert auf die Sicherheit ihrer Anwendung und ihrer Daten legen und dadurch eine		Der Quellcode des Prototyps wurde unter der GPL und das KiezRadar-Backend unter der LGPL veröffentlicht. https://gitlab.com/kiezradar

#	User Story	Akzeptanzkriterien	Machbarkeit
	Nachnutzung des Codes möglich ist.		

Tabelle 11: Nicht umgesetzte User Stories aus dem Epic »Verfügbarkeit und Installation«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E01.03	Die datenkritischen Nutzer:innen möchten die KiezRadar-App auch auf alternativen App-Stores (neben Google Play und Apple Store) wie bspw. F-Droid downloaden können, um nicht von Google oder Apple abhängig zu sein und die Open Source-Community zu unterstützen.		Ein Prototyp ist als Android-App umgesetzt. Eine Integration in den F-Droid App-Store ist demnach möglich.
E01.04	Die Nutzer:innen bevorzugen eine Webapp, da diese unabhängig von der Plattform (Betriebssystem, Gerät) genutzt werden kann und einen höheren Funktionsumfang erlaubt.		Nach Abwägung verschiedener Parameter wurde sich entschieden, den Prototypen als Android-App umzusetzen. Entsprechend ist das Design auf die Besonderheiten mobiler Endgeräte, wie bspw. deren geringe Bildschirmgröße abgestimmt. Eine Adaption als Webanwendung bei gleichzeitiger Anpassung des Designkonzepts ist aber möglich.

4.2.2 Informationsmenge steuern

Tabelle 12: Umgesetzte User Stories aus dem Epic »Informationsmenge steuern«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E02.01	Die Nutzer:innen möchten selbst einstellen können, wie ausführlich die Informationen angezeigt werden, um ihren individuellen Informationsbedürfnissen gerecht zu werden.	Die App erlaubt in einem sinnvollen Rahmen das Ein- und Ausklappen von Informationen. Wenn möglich, wird Information zusammengefasst dargestellt, mit der Option weitere Informationen anzuschauen.	Ist in der App umgesetzt.
E02.02	Die Nutzer:innen möchten eine reduzierte Informationsmenge angezeigt bekommen, die mit einem Klick auf »Details« eine Detailansicht ermöglicht, um einen	Wenn möglich, wird Information reduziert dargestellt mit der Option weitere Informationen anzuschauen.	Ist in der App durch ausklappbare Menüs umgesetzt.

#	User Story	Akzeptanzkriterien	Machbarkeit
	besseren und schnelleren Überblick zu bekommen.		
E02.03	Die Nutzer:innen möchten bei Themen, die sie als besonders interessant erachten, Verlinkungen zu weiterführenden Informationen vorfinden, um den Aufwand einer eigenen Recherche zu vermeiden.	Bei Einträgen gibt es einen Button, der auf eine Seite mit weiterführenden Informationen weiterleitet.	Ist in der App in der Einzelansicht von Einträgen umgesetzt.
E02.04	Die Nutzer:innen möchten die angezeigten Informationen personalisieren können, um ihren individuellen Bedürfnissen gerecht zu werden.	Durch Filtereinstellungen können Inhalte dauerhaft auf vorgegebene Parameter begrenzt werden. Filtereinstellungen können gespeichert werden, um ein erneutes Einstellen zu vermeiden.	Inhalte sind erarbeitet und in das Konzept, sowie in das Design (Mockups) eingebunden. Die Umsetzung in der App steht jedoch noch aus.
E02.05	Nutzer:innen möchten auf einen Blick sehen, welche Veranstaltungen und Beteiligungen im Moment verfügbar sind.	Einträge enthalten die Information, ob sie momentan verfügbar sind. Es ist einstellbar, dass momentan verfügbare Einträge dargestellt werden.	Ist in der App durch zeitliche Filterung umgesetzt.
E02.06	Nutzer:innen möchten eine Funktion, in die sie Filtereinstellungen übernehmen können und zusätzlich noch zwischen Einträgen in der Nähe und Ergebnissen, die aktuell sind wählen können.	Es gibt eine Funktion in der zwischen aktuellen Einträgen und Einträgen in der Nähe gewechselt werden kann, während andere Filterparameter beibehalten werden.	Ist in der App durch die Entdecken-Funktion umgesetzt.
E02.07	Nutzer:innen möchten alle Filterparameter auf einen Blick sehen, wenn sich das Filtermenü öffnet, damit sie wissen, welche Filterparameter es gibt.	Im Filtermenü sind alle Filterparameter auf einen Blick sichtbar.	Ist in der App durch ein ausklappbares Filtermenü umgesetzt.
E02.08	Nutzer:innen wünschen sich, Filterparameter durch eine direkte Texteingabe setzen zu können, um nicht im Menü danach suchen zu müssen.	Filter können durch eine direkte Texteingabe gesetzt werden.	Ist in der App umgesetzt.
E02.09	Nutzer:innen wünschen sich, dass beim Einstellen der Filter Schlagworte zu Themen und Kategorien weiter oben stehen, weil Inhalte wichtig sind.	Im Filtermenü befinden sich Schlagworte zu Themen und Kategorien oben.	Ist in der App umgesetzt.

4.2.3 Informationen zu Beteiligungsformaten und Veranstaltungen

Tabelle 13: Umgesetzte User Stories aus dem Epic »Informationen zu Beteiligungsformaten und Veranstaltungen«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E03.01	Die Nutzer:innen möchten auf aktuelle Daten und Informationen zugreifen, um Frust zu vermeiden und einen Anreiz für eine stete Nutzung zu haben.	Die App stellt aktuelle Informationen zur Verfügung.	Ist in der App durch Aktualität der Informationen im API umgesetzt.
E03.02	Nutzer:innen möchten auf einen Blick sehen, welche Veranstaltungen und Beteiligungen es in ihrer Nähe gibt.	Die App stellt eine Funktion bereit, in der Veranstaltungen und Beteiligungen in der Nähe angezeigt werden.	Ist in der App über den Reiter "In der Nähe" innerhalb der Entdecken-Funktion umgesetzt.
E03.03	Nutzer:innen möchten in der Listenansicht von Veranstaltungen und Beteiligungen weitere Informationen, wie den Ort, den Zeitpunkt und die Art des Events, sehen, um die Einträge besser für sich einordnen zu können.	Die Listenansicht zeigt wichtige Eintragsinformationen wie Ort, Zeitpunkt und Kategorie an.	Ist in der App umgesetzt.

Tabelle 14: Nicht umgesetzte User Stories aus dem Epic »Informationen zu Beteiligungsformaten und Veranstaltungen«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E03.04	Die Nutzer:innen möchten, dass die App für die Vielschichtigkeit von Beteiligungsprojekten sensibilisiert, um falschen Erwartungen vorzubeugen.	Nutzer:innen möchten detaillierte Projektinformationen, um den Prozess nachzuvollziehen.	Machbar: da Beteiligungskonzepte vorab erstellt werden müssen, muss von Anfang an feststehen, inwiefern Informationen genutzt werden und veröffentlicht werden. Dies ist nicht Aufgabe der KiezRadar-App, sondern der Datenanbieter. Daher wird es vorgezogen auf die entsprechenden Plattformen zu verlinken, statt detaillierte Informationen in der App vorzuhalten.
E03.05	Die Nutzer:innen und Nutzer möchten voll-umfassende Informationen zu Beteiligungsverfahren und Projekten erhalten, da Transparenz für Interesse und Vertrauen sorgt.	Einträge zu Beteiligungsverfahren und Projekten beinhalten voll-umfassende Informationen.	Die App soll Nutzer:innen vor Allem zu Informationsquellen führen. Es wird deshalb davon abgesehen, die gesamten Informationen in die App zu integrieren und auf die entsprechenden Webseiten verwiesen.
E03.06	Nutzer:innen wünschen sich, über Bauarbeiten bei der BVG informiert zu werden.	Bauarbeiten bei der BVG sind als Einträge in der KiezRadar-App verfügbar.	Die KiezRadar-App bezieht derzeit keine Daten der BVG.

#	User Story	Akzeptanzkriterien	Machbarkeit
E03.07	Nutzer:innen möchten die Möglichkeit haben, selbst Einträge erstellen zu können.	Nutzer:innen können selber Einträge, wie Veranstaltungen oder Beteiligungsprojekte erstellen.	Dies ist nicht Teil des Konzepts der App.
E03.08	Die Nutzer:innen möchten ein langfristiges und qualitativ anspruchsvolles Redaktionsmodell für die Betreuung der App, um die App langfristig nutzen zu können und zu einer steten Nutzung motiviert zu werden.	Der Inhalt der App wird redaktionell betreut. Die Betreuung ist langfristig gesichert.	Die App soll kein Redaktionssystem ersetzen, sondern vorhandene Datenquellen aufbereiten. Daher wird diese Anforderung nicht umgesetzt, obwohl sie reizvoll ist.

4.2.4 Kartenfunktion und standortbezogene Nutzung

Tabelle 15: Umgesetzte User Stories aus dem Epic »Kartenfunktion und standortbezogene Nutzung«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E04.01	Nutzer:innen möchten standortbezogene Information beziehen.	Die Karte setzt einen Marker für den aktuellen Standort.	Ist in der App umgesetzt, wenn die Nutzer:innen den eigenen Standort für die App freigegeben haben.
E04.02	Nutzer:innen möchten den Info-Radius einstellen können.	Der Radius zur ortsbezogenen Suche in der Karte kann eingestellt werden.	Ist in der App umgesetzt.
E04.03	Nutzer:innen möchten den Suchradius über eine einfach einstellbare Funktion eingeben können.	Der Suchradius kann durch Texteingabe oder einen Slider eingestellt werden.	Ist in der App umgesetzt.
E04.04	Nutzer:innen wünschen sich, dass eine ortsbezogene Suche sowohl für den aktuellen Standort, als auch für eine eingegebene Adresse durchgeführt werden kann.	Die ortsbezogene Suche kann sowohl für den aktuellen Standort, als auch für eine eingegebene Adresse durchgeführt werden.	Ist in der App umgesetzt.
E04.05	Nutzer:innen wünschen sich, dass der eigene Standort in der Karte markiert ist.	Der eigene Standort ist in der Karte markiert.	Diese Funktion kann durch einen Button in der Kartenansicht aktiviert werden und wurde damit in der App umgesetzt.
E04.06	Nutzer:innen wünschen sich, auch in der Listenansicht den Suchradius beschränken zu können.	Auch Einträge in der Listenansicht können auf einen bestimmten Suchradius begrenzt werden.	Ist in der App durch den Radius-Filter umgesetzt.
E04.07	Nutzer:innen wünschen sich über die Eingabe eines Radars durch Projekte und Veranstaltungen navigieren zu können.	Mit einer Radiusuche können Einträge innerhalb eines vorgegebenen Radius gefiltert werden.	Ist in der App umgesetzt.

Tabelle 16: Nicht umgesetzte User Stories aus dem Epic »Kartenfunktion und standortbezogene Nutzung«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E04.08	Nutzer:innen möchten einsehen, an welchen Projekten sie vorbeigefahren sind.	Über eine Funktion in der App können Nutzer:innen verfolgen, welche Beteiligungen und Veranstaltungen entlang einer von ihnen zurück gelegten Route bestehen.	Machbar, wird derzeit jedoch nicht umgesetzt.
E04.09	Nutzer:innen wünschen sich eine Kartensuche durch Einzeichnen eines Bereichs oder ziehen eines Radius.	Durch Einzeichnen eines Bereichs oder Ziehen eines Suchradius kann in der Karte die Suche begrenzt werden.	Das Einstellen eines Suchradius ist durch einen Slider in den Filtereinstellungen umgesetzt. Ein Markieren von Bereichen in der Karte, in denen gesucht wird, ist derzeit nicht umgesetzt.
E04.10	Nutzer:innen wünschen sich ein Vorschauenfenster für Einträge in der Karte.	Beim Anklicken oder Hovern über einen Eintrag in der Karte wird ein Vorschauenfenster des Eintrags angezeigt.	Im Prototypen aktuell nicht umgesetzt.

4.2.5 Beteiligungen wahrnehmen

Tabelle 17: Nicht umgesetzte User Stories aus dem Epic »Beteiligung wahrnehmen«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E05.01	Die Nutzer:innen möchten sich auch durch direkte Abstimmungen beteiligen, um Zeit und Aufwand zu sparen.	In der App kann an Beteiligungen teilgenommen werden.	Nicht sinnvoll als Teil der App, da die Repräsentativität des Ergebnisses und der Zugang zu einer solchen Beteiligung problematisch sind.
E05.02	Die Nutzer:innen möchten sich aus der App heraus beteiligen können, in dem sie über eine Auswahl mehrerer vorgegebener Alternativen abstimmen können, um Zeit und Aufwand zu sparen.	Auswahl mehrerer vorgegebener Alternativen wird in der KiezRadar-App zum Voting bereitgestellt.	Die App soll Nutzer:innen vor Allem zu Informationen führen. Es wird deshalb davon abgesehen, eigene Abstimmungen zu integrieren.
E05.03	Die Nutzer:innen möchten sich aus der App heraus beteiligen können, indem mittels der App Vorschläge und Handlungsalternativen erhoben werden, um Zeit und Aufwand zu sparen.	Über eine Funktion in der App können Ideen und Vorschläge eingereicht werden.	Die App soll Nutzer:innen vor Allem zu Informationen führen. Es wird deshalb davon abgesehen, Ideen-einreichungen zu integrieren.

4.2.6 Projektergebnisse verfolgen

Tabelle 18: Umgesetzte User Stories aus dem Epic »Projektergebnisse verfolgen«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E06.01	Nutzer:innen möchten Projekte und Veranstaltungen, die für sie interessant sind speichern.	Einträge können bspw. durch eine Favoritenliste gespeichert werden.	Ist in der App umgesetzt.
E06.02	Nutzer:innen möchten aus der Listenansicht heraus Einträge zu ihren Favoriten hinzufügen können.	In der Listenansicht sind Einträge mit einem Button zum Favorisieren versehen.	Ist in der App umgesetzt.
E06.03	Nutzer:innen möchten, dass die Favoriten durch einen gut sichtbaren Button geöffnet werden, ohne sich durch ein Menü klicken zu müssen.	Favoriten können durch einen dauerhaft sichtbaren Button geöffnet werden, beispielsweise als Teil einer Fußleiste.	Ist in der App umgesetzt.

Tabelle 19: Nicht umgesetzte User Stories aus dem Epic »Projektergebnisse verfolgen«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E06.04	Die Nutzer:innen möchten über Veränderungen oder Entwicklungen von bestimmten Projekten oder Themenbereichen informiert werden, um Entwicklungen und Ergebnisse nachvollziehen zu können.	Die Stationen eines Projektes werden ebenso wie die geplanten Punkte an denen Ergebnisse veröffentlicht werden sollen, vorab definiert und transparent auf den unter »Details« verlinkten Webseiten dargestellt.	In den Berliner Leitlinien für Bürgerbeteiligung formuliert der Punkt 7 ein grundsätzliches Streben danach, wichtige Informationen in einer Vorhabenliste zu veröffentlichen und unter Nutzung verschiedener Kanäle (u.a. der zentralen Beteiligungsplattform) zu veröffentlichen. Verbindlicher formuliert der Punkt 8 eine Rechenschaftspflicht in Bezug auf die Inhalte die durch Bürger:innen eingebracht wurden. Hier muss eine schriftliche und öffentliche Stellungnahme zu den Empfehlungen und Anforderungen der Bürger:innen erfolgen. Die genauen Informationspflichten, Kommunikationskanäle und Abläufe werden für jede Beteiligung individuell innerhalb des jeweiligen Beteiligungskonzepts festgelegt. Teilnehmende des Verwaltungsworkshops weisen indes daraufhin, dass diese in der Praxis aufgrund der Personalknappheit nicht immer erfüllt werden können.

#	User Story	Akzeptanzkriterien	Machbarkeit
E06.05	Die Nutzer:innen möchten rechtzeitig vor dem Ablauf bestimmter Fristen für eine Veranstaltung informiert werden, um sich rechtzeitig beteiligen zu können.	N Tage vor dem Ablauf der Fristen wird auf den Ablauf von Veranstaltungen hingewiesen.	In der Regel stehen die Zeiträume bzw. die Daten für das Stattfinden von Beteiligungsmöglichkeiten als Datensatz zur Verfügung, so dass diese Anforderung umgesetzt werden kann. Auch das Einschränken von Erinnerungen (z. B. keine oder nur abonnierte) kann unter "Einstellungen" vorgenommen werden.
E06.06	Nutzer:innen möchten den Zusammenhang zwischen Ergebnissen und Projektschritten über den gesamten Projektverlauf nachvollziehen.	Nicht nur das jeweilige Projekt wird in den Schritten abgebildet, sondern auch vorausgegangene Sitzungen etc. werden zusammen mit dem Projekt angezeigt.	Je nach Quelle stehen diese Informationen zur Verfügung oder nicht. In jedem Fall muss in der App oder dem Backend die Information vorgehalten werden, auch wenn z. B. Projektschritte gelöscht werden. Das ist machbar, ist aber sehr aufwendig in all seinen Konsequenzen.
E06.07	Nutzer:innen möchten Verfahren abonnieren können, in der Form, dass sie über die gesamte Entwicklung, auch über zusammenhängende Ereignisse, informiert werden.	Nutzer:innen können Einträge abonnieren.	Events können in den Favoriten gespeichert werden. Ein Benachrichtigungskonzept für bestimmte Filter und Favoriten-Einstellungen ist konzeptionell ausgearbeitet und in das Design (Mockups) eingebunden. Die Verlinkung im Prototypen steht allerdings noch aus. In beide Fällen können auch zukünftig verknüpfte Events darüber abgebildet werden. Die derzeitige Datenlage ermöglicht jedoch noch keine Verknüpfung von Events.
E06.08	Nutzer:innen wünschen sich eine Aufteilung der Benachrichtigungen nach Kategorien.	Die Übersicht über Benachrichtigungen ist nach Kategorien aufgeteilt.	Machbar, wird jedoch derzeit nicht umgesetzt und ist nicht von allen Nutzer:innen gewünscht.

4.2.7 Interoperabilität/ Integration

Tabelle 20: Umgesetzte User Stories aus dem Epic »Interoperabilität/ Integration«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E07.01	Die Nutzer:innen möchten, dass die App eine Vielzahl an Medien und Inhalten von anderen Plattformen einbindet, damit sie eine möglichst breite Informationsbasis zu Veranstaltungsangeboten erhalten.	Die App stellt Inhalt aus verschiedenen Plattformen und Medien zur Verfügung.	Es werden verschiedene Datenquellen genutzt. Sofern die Daten im genutzten Format in weiteren Datenquellen zur Verfügung stehen, ist eine Erweiterung möglich.

Tabelle 21: Nicht umgesetzte User Stories aus dem Epic »Interoperabilität/ Integration«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E07.02	Die Nutzer:innen wünschen sich eine Integration in eine weitere bereits verbreitete App, um die Reichweite und Nutzergruppe der KiezRadar-App zu vergrößern.	Die App steht als Unterfunktion von verbreiteten kommunalen Apps zu Verfügung, bspw. BVG-App oder Berlin Service-App.	Die KiezRadar-App wird derzeit als eigenständige App entwickelt. Eine Integration der App ist nicht geplant. Das Backend und dessen API sind jedoch offen. Sie können von anderen Apps oder Anwendungen genutzt werden.
E07.03	Die Nutzer:innen möchten, dass die KiezRadar-App die Integration anderer App-Services zulässt, u.a. das Teilen von Inhalten oder das Importieren externer Inhalte, um eine effiziente Einbindung von Informationen zu ermöglichen.	Einträge können auf andere Plattformen durch eine »Teilen«-Funktion exportiert werden. Termine können in Kalender exportiert werden.	Theoretisch wäre das Anbieten vom Exportieren der Inhalte möglich. Dies wird derzeit noch nicht umgesetzt. Eine direkte Integration von Social Media Plattformen gestaltet sich schwieriger. Integrations- und Teilen-Funktionen sind sowohl von der Nutzung der Daten als auch vom Datenschutz her zu bedenken und zu konzipieren. Das ist im Rahmen des Prototypen nicht machbar.

4.2.8 Kommunikation

Tabelle 22: Nicht umgesetzte User Stories aus dem Epic »Kommunikation«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E08.01	Die Nutzer:innen sehen die KiezRadar-App nicht nur als Informationsquelle, sondern möchten auch aktiv mit anderen Nutzer:innen der App in Kontakt treten, um sich zu vernetzen.	Eine Kontaktaufnahme mit anderen Nutzer:innen ist durch die App möglich.	Dies ist nicht Teil des Konzepts der App.
E08.02	Die Nutzer:innen möchten miteinander interagieren, indem sie sich spontan vernetzen durch Initiativen und indem sie diese starten, ihnen beitreten oder sich über sie informieren können.	In der App können Initiativen gegründet werden, denen Nutzer:innen beitreten oder über die sie sich informieren können.	Dies ist nicht Teil des Konzepts der App.
E08.03	Die Nutzer:innen möchten gerne über die KiezRadar-App mit der Berliner Verwaltung direkt in Kontakt treten können.	In der App kann mit der Verwaltung Kontakt aufgenommen werden durch das Angeben einer Mailadresse oder ein Kontaktformular.	Dies ist nicht Teil des Konzepts der App.

#	User Story	Akzeptanzkriterien	Machbarkeit
E08.04	Nutzer:innen möchten Kommentare bei Einträgen einfügen können, die für alle sichtbar sind und sich mit anderen Nutzer:innen vernetzen können.	Nutzer:innen können zu Einträgen Kommentare machen, die für alle Nutzer:innen sichtbar sind und andere Nutzer:innen kontaktieren.	Dies ist nicht Teil des Konzepts der App und würde mit dem derzeitigen Datenschutzkonzept nicht einhergehen.

4.3 Supportfunktionen

4.3.1 Hilfe und Datenschutz

Tabelle 23: Umgesetzte User Stories aus dem Epic »Hilfe und Datenschutz«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E09.01	Die Nutzer:innen wünschen sich die Möglichkeit die KiezRadar-App anonym nutzen zu können, um keine Daten weitergeben zu müssen.	Die App erhebt und verarbeitet keine personenbezogenen Daten. Falls Funktionen personenbezogene Daten benötigen, können diese deaktiviert werden.	Ist in der App umgesetzt. Personenbezogene Daten werden ausschließlich auf dem Handy gespeichert. Es werden keine Daten verarbeitet.
E09.02	Die Nutzer:innen möchten nicht, dass mehr Berechtigungen und Datenzugriffen vorgegeben werden als es für die Funktionen notwendig ist, um die App möglichst datensparsam nutzen zu können.	Es werden nur Daten verarbeitet, die für die Funktionalität der App nötig sind.	Ist in der App umgesetzt.
E09.03	Die Nutzer:innen möchten eine Alternative zu einer standortbezogenen Suche angeboten bekommen, um selbst über die Preisgabe der Daten bestimmen zu können.	Der Standort muss nicht angegeben werden, um eine ortsbezogene Suche durchzuführen.	Ist in der App umgesetzt.
E09.04	Die Nutzer:innen wünschen sich eine Offenlegung des Quellcodes der KiezRadar-App, um Vertrauen in die Anwendung aufzubauen.	Der Quellcode ist vollständig und zugänglich verfügbar.	Wird umgesetzt.
E09.05	Die Nutzer:innen wünschen sich eine Offenlegung möglichst vieler Projektinformationen, um die Entwicklung des Projektes nachverfolgen zu können.	Innerhalb der App sind alle nötigen Informationen, um das Projekt zu verstehen verfügbar. Durch eine Weiterleitung auf die Projektseite können Nutzer:innen die gesamten Informationen zum Projekt abrufen.	Die Projektseite ist online und bildet den aktuellen Stand des Projekts ab, siehe https://kiezradar.fokus.fraunhofer.de/ . Demnach sind die Inhalte erarbeitet und in das Konzept, sowie das Design (Mockups) eingebunden. Die Verlinkung im Prototypen steht allerdings noch aus.

#	User Story	Akzeptanzkriterien	Machbarkeit
E09.06	Die Nutzer:innen wünschen sich eine Offenlegung der bei der vom KiezRadar genutzten Daten, um sie für eigene Zwecke nutzen zu können.	Die in der App genutzten Daten stehen über eine öffentliche API zur Verfügung.	Ist mit der KiezRadar-API umgesetzt, siehe https://kiezradar.fokus.fraunhofer.de/api/ .
E09.07	Die Nutzer:innen und Nutzer möchte klare Informationen zu den Herstellern der App, um der App Vertrauen entgegen zu bringen.	Die App beinhaltet ein Impressum in dem eingesehen werden kann, wer für die Entwicklung der App verantwortlich ist.	Ist über Gitlab und die Webseite nachlesbar. Die Integration in der App steht jedoch noch aus.

Tabelle 24: Nicht umgesetzte User Stories aus dem Epic »Hilfe und Datenschutz«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E09.08	Die Nutzer:innen möchten, dass die Themen Anonymität und Datensicherheit besonders thematisiert werden, damit sie der App Vertrauen entgegenbringen können.	Bei erstmaligem Ausführen der App wird über Datenschutzbestimmungen und Nutzungsbedingungen informiert und die Zustimmung von Nutzer:innen eingeholt. Die Datenschutzbestimmungen und Nutzungsbedingungen sind in der App dauerhaft einsehbar.	Inhalte erarbeitet und in das Konzept, sowie das Design (Mockups) eingebunden. Die Umsetzung im Prototypen steht allerdings noch aus.
E09.09	Die Nutzer:innen möchten ausführlich darüber informiert werden, wann (bei welchen Funktionen), zu welchem Zweck, von wem und warum, welche Daten erhoben werden, um der App Vertrauen entgegen zu bringen.	Eine dauerhaft einsehbare Datenschutzhinweise gibt verständlich anhand von Beispielen Auskunft darüber, wie durch die App personenbezogene Daten erhoben werden und zu welchem Zweck.	Inhalte erarbeitet und in das Konzept, sowie das Design (Mockups) eingebunden. Die Umsetzung im Prototypen steht allerdings noch aus.
E09.10	Die Nutzer:innen möchten funktionsbezogen über die Preisgabe ihrer Daten informiert und bestimmen können, um in der Lage zu sein, selbstbestimmt über die Weitergabe ihrer Daten verfügen zu können.	Bei erstmaligem Ausführen der App wird über Datenschutzbestimmungen und Nutzungsbedingungen informiert und die Zustimmung von Nutzer:innen eingeholt. Die Datenschutzbestimmungen und Nutzungsbedingungen sind in der App dauerhaft einsehbar.	Inhalte erarbeitet und in das Konzept, sowie das Design (Mockups) eingebunden. Die Umsetzung im Prototypen steht allerdings noch aus.

#	User Story	Akzeptanzkriterien	Machbarkeit
E09.11	Die Nutzer:innen möchten eine App verwenden, bei der verschiedene Formen des Missbrauchs aktiv durch Hilfen und Hinweise adressiert werden, um eine konfliktfreie Nutzung der App zu erfahren.	Es besteht die Möglichkeit, Einträge und Nutzer:innen zu melden.	Eine Moderation von Inhalten, um aktiv auf Missbrauch hinzuweisen ist nötig. Dies ist bisher nicht vorgesehen.
E09.12	Die Nutzer:innen möchten, dass von Anfang an möglichst konkret kommuniziert wird, was mit den Ergebnissen einer Veranstaltung oder Beteiligung passiert, um sicher zu sein, dass diese auch Einfluss nehmen.	Bevor Nutzer:innen an einer Veranstaltung oder Beteiligung teilnehmen werden sie darüber informiert, wie Daten verwendet werden.	Machbar, da Beteiligungskonzepte vorab erstellt werden müssen, muss von Anfang an feststehen, inwiefern Informationen genutzt werden und veröffentlicht werden. Dies ist nicht Aufgabe der KiezRadar-App, sondern der Datenanbieter.

4.3.2 Einstellungsmöglichkeiten

Tabelle 25: Umgesetzte User Stories aus dem Epic »Einstellungsmöglichkeiten«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E10.01	Die Nutzer:innen möchten die Benachrichtigungen genau einstellen können (was, wann, wie viele), um nicht von der Anwendung genervt zu sein.	Die Einstellung von Benachrichtigungen ist möglich in Bezug auf Informationsgehalt und Frequenz.	Eine Einstellungsmöglichkeit der Benachrichtigungen war für den Prototypen angedacht, konnte aber aus Zeitgründen nicht implementiert werden.
E10.02	Nutzer:innen wünschen sich beim erstmaligen Starten der App nach ihren Interessen gefragt zu werden, damit diese als Filterparameter übernommen werden.	Nach der Installation werden Nutzer:innen beim erstmaligen Öffnen der App nach ihren Interessen gefragt, die als Filterparameter übernommen werden.	Machbar, wird jedoch derzeit nicht umgesetzt.

4.3.3 Barrierefreiheit

Tabelle 26: Umgesetzte User Stories aus dem Epic »Barrierefreiheit«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E11.01	Nutzer:innen mit Farbschwächen wünschen sich, dass Farben nicht ausschließlich verwendet werden, um eine Bedeutung zu transportieren, damit sie die Bedeutung auch verstehen können.	Die App enthält keine Information, die ausschließlich über Farbe dargestellt wird.	Ist in der App umgesetzt.
E10.02	Nutzer:innen mit einer Sehschwäche oder Screenreadern wünschen sich zusätzlich zur Kartendarstellung eine textuelle	Alle Kartenmarker sind mit Adresse angegeben.	Ist in der App umgesetzt. Eine Überprüfung wurde indes nicht durchgeführt.

#	User Story	Akzeptanzkriterien	Machbarkeit
	Beschreibung der Marker, um diese erfassen zu können.		
E10.03	Nutzer:innen mit einer Sehschwäche oder Screenreadern wünschen sich, dass falls die Karte dazu dient, mehrere Standorte anzuzeigen, dass dann zusätzlich zur Karte eine Liste der Standorte angeboten wird, um sicherzustellen, dass sie diese erfassen können.	Zusätzlich zu Einträgen auf einer Karte, sind die Einträge als Liste verfügbar.	Ist in der App umgesetzt. Eine Überprüfung wurde indes nicht durchgeführt.

Tabelle 27: Nicht umgesetzte User Stories aus dem Epic »Barrierefreiheit«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E11.04	Die Nutzer:innen die einen Screenreader benutzen, wünschen sich eine Bedienung über die Standard-Wischgesten von Screen-Readern bzw. über Tastatursteuerung anderer assistiver Technologien (z. B. mobile Braillezeile), um die App nutzen zu können.	Die App kann mit einem Screen-Reader über Standard-Wischgesten bedient werden oder über die Tastatursteuerung anderer assistiver Technologien.	Machbar, wird derzeit jedoch nicht umgesetzt.

4.3.4 Navigation und Bedienung (Usability)

Tabelle 28: Umgesetzte User Stories aus dem Epic »Navigation und Bedienung (Usability)«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E12.01	Die Nutzer:innen wünschen sich, dass möglichst kein horizontales Scrollen notwendig ist, da dies bei Smartphones umständlich ist.	Die App ist so gestaltet, dass horizontales Scrollen nicht nötig ist.	Ist in der App umgesetzt.
E12.02	Die Nutzer:innen wünschen sich einen Mechanismus um Aktionen wieder rückgängig zu machen oder die Aktion abubrechen, um nicht gewollte Aktionen zurückzunehmen.	Interaktionsdialoge in der App können abgebrochen werden. Eingegebene Inhalte können gelöscht werden.	Ist in der App umgesetzt.
E12.03	Die Nutzer:innen wünschen sich, dass Tastatureingaben möglichst vermieden werden, um Fehler in der Eingabe zu vermeiden und Zeit zu sparen.	Wenn möglich, gibt es Vorschläge oder eine Auswahl, um eine Tastatureingabe zu vermeiden.	Ist in der App umgesetzt.

#	User Story	Akzeptanzkriterien	Machbarkeit
E12.04	Die Nutzer:innen wünschen sich, falls eine Eingabe vorgenommen werden soll, dass die für die Eingabe passende Tastatur angezeigt wird (Zahlen, Text, E-Mail), für eine möglichst bequeme Eingabe.	Bei Eingaben wird die Tastatur angezeigt, die dem Inhalt der Eingabe entspricht.	Ist in der App umgesetzt.
E12.05	Die Nutzer:innen wünschen sich, dass bei der Verwendung von Schaltflächen die wichtigen bzw. zu erwartenden Schaltflächen hervorgehoben werden, um eine schnelle Bedienbarkeit zu ermöglichen.	Wichtige bzw. zu erwartende Schaltflächen sind hervorgehoben.	Ist in der App umgesetzt.
E12.06	Die Nutzer:innen wünschen sich, dass sie die Größe von Karten mittels Pinching verändern können, um eine intuitive Bedienbarkeit zu erzielen.	Die Karte kann mittels Pinching verkleinert oder vergrößert werden.	Ist in der App umgesetzt.
E12.07	Die Nutzer:innen möchten, dass das Drücken auf Text oder Icons zum selben Ergebnis führt, um eine schnelle Bedienung zu ermöglichen.	Sowohl Text als auch zugehörige Icons können gedrückt werden, um die Aktion auszulösen.	Ist in der App umgesetzt.
E12.08	Die Nutzer:innen wollen nicht zu viele Einstellungen vornehmen müssen oder diese eher im Laufe der Nutzung integriert einstellen, um sofort mit der Nutzung starten zu können.	Die App lässt sich auch ohne persönliche Einstellungen nutzen. Ohne persönliche Einstellungen werden alle Inhalte angezeigt.	Ist in der App umgesetzt.
E12.09	Die Nutzer:innen wollen eine App, die bei der Bedienbarkeit nicht zu viele und nur bereits bekannte Gesten verwendet, da diese erst entdeckt werden müssen.	Die App wird mehrheitlich durch bekannte Gesten bedient.	Ist in der App umgesetzt.
E12.10	Die Nutzer:innen wollen eine App, die möglichst schnell und performant ist, um Zeit zu sparen.	Die App reagiert schnell und zeigt neu geöffnete Seiten schnell an.	Ist in der App umgesetzt.
E12.11	Die Nutzer:innen wollen eine App, die ein Problem/Aufgabe möglichst gut und einfach löst, um überflüssige Schritte und Funktionen zu vermeiden.	Nutzer:innen gelangen ohne Umwege zu den Informationen, die sie benötigen.	Ist in der App umgesetzt.

#	User Story	Akzeptanzkriterien	Machbarkeit
E12.12	Die Nutzer:innen möchten eine App die nicht mit Inhalten und Funktionen überfrachtet ist, für eine möglichst effiziente Nutzung und geringe Einarbeitungszeit.	Die App enthält nur Inhalte, die für die Information über Beteiligungsverfahren und Veranstaltungen im Kiez relevant sind.	Ist in der App umgesetzt.
E12.13	Nutzer:innen wollen eine App, die direkt nach der Installation nutzbar ist, um möglichst wenig Zeit aufwenden zu müssen.	Die App kann nach der Installation geöffnet und genutzt werden, ohne einen zeitaufwendigen Verifizierungsprozess o. Ä. durchführen zu müssen.	Ist in der App umgesetzt.
E12.14	Die Nutzer:innen möchten durch ein ansprechendes und originelles Design angesprochen werden, damit sie Spaß bei der Bedienung der App empfinden.	Die App hat ein ansprechendes und originelles Design.	Ist in der App umgesetzt. Das Feedback zum Design ist in den Nutzertests sehr positiv ausgefallen. Hinweise zum Design der Platzhalterbilder und zu den Kartenmarkern wurden in einer neuen Version umgesetzt.
E12.14	Die Nutzer:innen möchten die App intuitiv bedienen können, damit die Gewöhnungszeit/ Einarbeitungszeit möglichst gering oder im Optimalfall nicht vorhanden ist.	Die App verfügt über eine intuitive Bedienung.	Ist in der App umgesetzt. Die Bedienung der App wurde in den Nutzertests mehrheitlich intuitiv verstanden.
E12.15	Die Nutzer:innen möchten eine übersichtliche Gestaltung und Informationsmenge, um die Nutzung der App nicht als überfordernd zu empfinden.	Die App ist übersichtlich gestaltet und enthält eine angemessene Menge von Informationen.	Ist in der App umgesetzt. Die Informationsmenge der App wurde in den Nutzertests als angenehm empfunden.
E12.15	Die Nutzer:innen möchten, dass Informationen möglichst visuell, bspw. durch die Nutzung von Bildern vermittelt werden, um die Informationen schneller erfassen zu können.	Wenn möglich werden Informationen visuell übermittelt oder visuell unterstützt.	Ist in der App umgesetzt.
E12.16	Die Nutzer:innen möchten eine App im modernen Design, das sich von den gewohnten Verwaltungsstil abhebt, um offen und vorbehaltlos gegenüber einer Nutzung der App eingestellt zu sein.	Die App verfügt über ein modernes Design.	Ist in der App umgesetzt. Das Feedback zum Design ist in den Nutzertests sehr positiv ausgefallen.
E12.17	Die Nutzer:innen möchten, dass die Funktionen und das Design der App bezirksübergreifend einheitlich gestaltet sind, damit eine	Die App hat die gleichen Funktionen und das gleiche Design in allen Berliner Bezirken.	Ist in der App umgesetzt.

#	User Story	Akzeptanzkriterien	Machbarkeit
	bezirksübergreifende Zusammenarbeit funktioniert.		
E12.18	Die Nutzer:innen möchten eine effiziente Navigation, um die App möglichst zeitsparend zu nutzen.	Die Navigation ist effizient gestaltet.	Ist in der App umgesetzt. In den Nutzertests wurde die App Navigation mehrheitlich direkt verstanden und gut gehandhabt.
E12.19	Die Nutzer:innen wünschen sich große Bedienelemente, für eine leichte Bedienbarkeit.	Bedienelemente sind so groß gestaltet, wie möglich.	Ist in der App umgesetzt.
E12.20	Die Nutzer:innen wünschen sich dass die Tastatur der mobilen Endgeräte den Inhalt nicht zu sehr überdeckt, damit sie den Inhalt gut lesen können.	Die Tastatur zur Eingabe hat eine angemessene Größe.	Ist in der App umgesetzt.
E12.21	Die Nutzer:innen möchten die KiezRadar-App ohne Werbung nutzen, um nicht abgelenkt zu werden und das Gefühl zu haben, dass Daten für Werbezwecke gesammelt werden.	Die App enthält keine Werbung.	Ist in der App umgesetzt.
E12.22	Nutzer:innen möchten, dass Icons in der Fußleiste textuell beschrieben sind.	Icons in der Fußleiste sind textuell beschrieben.	Ist in der App umgesetzt.
E12.23	Nutzer:innen wünschen sich, dass Übersichten nicht mit Informationen überladen sind.	Übersichten enthalten eine reduzierte Menge von Informationen, die schnell von Nutzer:innen erfasst werden.	Ist in der App umgesetzt.
E12.24	Nutzer:innen wünschen sich eine klar verständliche Abgrenzung zwischen den Funktionen der App.	Funktionen in der App sind überschneidungsfrei und leicht verständlich.	Ist in der App umgesetzt.

Tabelle 29: Nicht umgesetzte User Stories aus dem Epic »Navigation und Bedienung (Usability)«.

#	User Story	Akzeptanzkriterien	Machbarkeit
E12.25	Nutzer:innen mit einer anderen Muttersprache als deutsch möchten die App in ihrer Muttersprache nutzen können, um ein möglichst gutes und tiefes Verständnis über die Inhalte zu entwickeln und diese schnell erfassen zu können.	Die App ist auf verschiedenen Sprachen verfügbar. Die Sprache ist in der App einstellbar.	Der Prototyp wurde auf Deutsch entwickelt, jedoch wurden die Android-Mechanismen genutzt, die eine einfache Übersetzung möglich machen. Diese ist im Prototyp nicht vorgesehen. Schwieriger wird es, die Inhalte, also Projektinformationen etc. zu übersetzen, da diese derzeit nur

#	User Story	Akzeptanzkriterien	Machbarkeit
			einsprachig vorliegen und automatisierte Übersetzungen zu Fehlern führen könnten.
E12.26	Die Nutzer:innen wollen, dass ein Drücken auf das Logo bzw. den Anwendungsnamen sie zurück zur Startseite führt, weil sie das so gewohnt sind.	Beim Drücken des KiezRadar-Icons in der App öffnet sich die Startseite.	Die eigentliche Einstiegsseite in die App ist die Seite mit einer Listenansicht aller oder der gefilterten Events. Ein Radar-Icon, das an das KiezRadar-Icon erinnert ist in der Navigation dieser Ansicht zugeordnet. Eine spezielle Platzierung des Kiezradar-Icons über alle Ansichten gibt es nicht.
E12.27	Die Nutzer:innen möchten ein variables, flexibles Design, damit die App viele verschiedene Nutzergruppen anspricht.	Das Design der App ist einstellbar.	Das Design kann nicht individualisiert werden. Mehrere Designs wie z.B ein dark-mode, wurden im Rahmen der Prototypen-Erstellung nicht umgesetzt, da bis zuletzt durch Tests noch Änderungen am Hauptdesign vorgenommen wurden.
E12.28	Die Nutzer:innen wünschen sich eine leicht auffindbare Möglichkeit sich von der App auszuloggen, um Vertrauen in die Anwendung zu haben und sicher zu sein, dass sie beendet ist.	Ein Log-Out ist möglich und gut sichtbar in die App integriert.	Es gibt kein Log-Out, da es kein Log-In gibt, die App kann lediglich wie alle anderen Apps geschlossen werden.
E12.29	Nutzer:innen wünschen sich, Termine von Projekten und Veranstaltungen in ihren persönlichen Terminkalender zu übernehmen.	Termine können in den Kalender von Nutzer:innen exportiert werden.	Machbar, wird derzeit jedoch nicht umgesetzt.
E12.30	Die Nutzer:innen möchten eine stabile App nutzen ohne Abstürze, um die App effizient zu nutzen und Datenverluste zu vermeiden.	Die App wird stabil und ohne Abstürze ausgeführt.	Die App ist ein Prototyp und daher nicht ausschließlich auf Stabilität ausgelegt. Ein Absturz kann jedoch keinen Datenverlust auslösen, da der Prototyp mit aktuellen Daten von der Schnittstelle zum Backend arbeitet.
E12.31	Nutzer:innen wünschen sich, dass ein "Entdecken" Icon dem Icon der KiezRadar-App entspricht.	Der "Entdecken" Icon entspricht dem Design des allgemeinen KiezRadar Icons.	Der Entdecken Icon ist derzeit als eigenständiges Icon designt.
E12.32	Nutzer:innen wünschen sich, dass Kacheln im Filter sichtbar sind, ohne scrollen zu müssen.	Kacheln, die die eingestellten Filterparameter anzeigen, sind ohne scrollen sichtbar.	Aufgrund der vielzähligen Filtermöglichkeiten und der begrenzten Bildschirmgröße ist eine Anzeige aller Filterparameter ohne

#	User Story	Akzeptanzkriterien	Machbarkeit
			Scrollen schlichtweg nicht möglich.

4.4 Anwendungsszenarien

Anwendungsszenarien dienen dazu, die angestrebte Interaktion zwischen den Nutzer:innen und dem System ins Zentrum zu rücken. Dadurch lassen sich die Interaktionsschritte bspw. kritisch betrachten oder veranschaulichen. Bei den Szenarien war uns wichtig, die drei Datenquellen, auf die wir uns beim Prototyp der KiezRadar-App fokussieren, jeweils mit einem Anwendungsfall abzubilden: mein.berlin.de, die Ratsinformationssysteme der Bezirke und den Veranstaltungskalender des Quartiersmanagement Auguste-Viktoria-Allee.

4.4.1 Beispiel Quartiersmanagement

Akteur:innen: Benutzer:innen (Aliye)

Auslöser: Aliye hat die Kiezradar-App frisch installiert und möchte für sie passende Angebote entdecken.

Normalablauf:

- 1 Aliye startet die KiezRadar-App und klickt auf die Filterfunktion oben rechts in der KiezRadar-App.
- 2 Dort wählt sie für sie passende Themenbereiche aus und setzt den Umkreis für die Anzeige von Einträgen auf einen Radius von 5 km um ihren Wohnort.
- 3 Aliye geht auf »Entdecken« und sieht sich die Einträge unter dem ersten Reiter »Eigene« an.
- 4 Der wöchentlich stattfindende Upcycling-Workshop weckt Aliyes Interesse und sie klickt auf die Veranstaltung für weitere Informationen.
- 5 Aliye liest sich die Beschreibung durch, fügt den nächsten Termin dem Smartphone-eigenen Kalender zu und markiert die Reihe unter Favoriten.

4.4.2 Beispiel mein.berlin.de

Akteur:innen: Benutzer:innen (Florian)

Auslöser: Florian erhält durch die KiezRadar-App eine Benachrichtigung über die Beteiligungsmöglichkeit »Aktionsfonds 2020 QM Flughafenstraße«. Die Beteiligung entspricht seinen eingestellten Präferenzen in Bezug auf das Wohngebiet und das Themengebiet »Stadtentwicklung«.

Normalablauf:

- 1 Florian sieht anhand des Benachrichtigungsicons, dass eine neue Benachrichtigung vorliegt und lässt sich diese anzeigen.
- 2 Durch das Klicken auf den Infotext zu der Beteiligungsmöglichkeit »Aktionsfonds 2020 QM Flughafenstraße« gelangt Florian zu dem Detailbereich der Beteiligung.
- 3 Im Detailbereich liest sich Florian die zusammenfassende Projektbeschreibung durch.

- 4 Florian klickt auf den Button »Beteiligen« und wird auf die externe Projektwebseite von »Aktionsfonds 2020 QM Flughafenstraße« auf mein.berlin.de.de weitergeleitet.
- 5 Florian bekommt eine Informationsseite zur Vorschlagsphase angezeigt und klickt auf den Button »Vorschlag anlegen«.
- 6 Florian gibt seine Nutzerdaten für mein.berlin.de ein.
- 7 Florian gibt einen Vorschlag für das Beteiligungsverfahren ab: er versieht seinen Vorschlag mit einem Titel, verfasst einen beschreibenden Text, lädt ein Bild hoch, bestätigt den Besitz der Urheberrechte an dem Bild, legt einen Budgetrahmen fest und verortet seinen Vorschlag durch die Angabe einer Straße und Ortsbezeichnung.
- 8 Florian schließt den Vorgang durch Klicken auf den Button »Speichern« ab.
- 9 Florian wechselt zurück in die KiezRadar-App und fügt das Projekt zu seinen Favoriten hinzu, um über die weiteren Projektfortschritte informiert zu werden.

Alternative Abläufe

- 3.1 Florian liest sich die Projektbeschreibung nicht in der KiezRadar-App durch, sondern auf der Projektseite von mein.berlin.de
- 5.1 Florian klickt auf den Button Anmelden/Registrieren und wählt bei den ausklappenden Reitern den Schritt »registrieren« aus.
- 5.2 In der sich öffnenden Maske gibt Florian seine E-Mail-Adresse ein und wählt einen Benutzernamen und ein Passwort. Nun wählt er aus, dass er keine Benachrichtigungen zu den Beteiligungen über die Mailadresse bekommen möchte und abonniert den Newsletter von mein.berlin.de. Anschließend willigt er in die Verarbeitung seiner Daten ein und bestätigt durch das Lösen eines Rätsels, dass er kein Roboter ist.

4.4.3 Beispiel Ratsinformationssystem

Akteur:innen: Benutzer:innen (Nora)

Auslöser: Nora interessiert sich für das Thema Clubkultur in Berlin und hat Sitzungen zu diesem Thema unter Favoriten gespeichert. Sie bekommt eine Benachrichtigung mit dem Datum der nächsten Sitzung angezeigt.

Normalablauf:

- 1 Nora sieht anhand des Benachrichtigungsicons, dass eine neue Benachrichtigung vorliegt und lässt sich diese anzeigen.
- 2 Durch das Klicken auf den Text der Benachrichtigung gelangt Nora zu dem Detailbereich der Tagesordnung der Sitzung und findet das Thema Clubkultur in Berlin unter Tagesordnungspunkt 9 »Open-Airs 2020 ermöglichen, geeignete Flächen ausweisen, Genehmigungsverfahren erleichtern, Clubkultur retten!«
- 3 Nora fügt den nächsten Sitzungstermin dem Smartphone-eigenen Kalender hinzu.

4.4.4 Beispiel Projekte und Veranstaltungen entlang einer Route

Akteur:innen: Benutzer:innen (Florian)

Auslöser: Florian möchte mit dem Fahrrad durch einen Berliner Bezirk fahren und interessiert sich dafür, welche Beteiligungsprojekte und Veranstaltungen entlang dieser Route existieren. Er schaltet deshalb die Streckennachverfolgung in der KiezRadar-App an, bevor er losfährt.

Normalablauf:

- 1 Florian öffnet die Filterfunktion in der KiezRadar-App.
- 2 Er klappt den Menüpunkt »Streckenverfolgung« aus.
- 3 Florian setzt den Toggle Switch von »aus« auf »an«. Neben dem Toggle Switch gibt er ein, in welcher Entfernung von der Strecke Projekte und Veranstaltungen gezeigt werden sollen. Eine Benachrichtigung erscheint im Menü des Betriebssystems seines Handys, die ihm mitteilt, dass die Streckenverfolgung durch KiezRadar aktiviert ist.
- 4 Florian fährt eine Strecke innerhalb von Berlin ab.
- 5 Nach der Fahrt deaktiviert Florian die Streckenverfolgung, indem er den Toggle Switch auf »aus« setzt. Die App benachrichtigt ihn, dass die Route gespeichert wurde.
- 6 Unterhalb des Toggle Switches erscheint ein Listeneintrag mit der abgefahrenen Route als Karten- oder Listenversion.
- 7 Florian drückt auf die Kartenversion.
- 8 Die App wechselt in die Kartenansicht und zeigt die von Florian abgefahrte Route zusammen mit Projekten und Veranstaltungen, die entsprechend der vorher eingegebenen Entfernung von der Route entfernt sind.
- 9 Florian schaut sich die Einträge an und löscht danach die eingespeicherte Route.

Alternative Abläufe

- 7.1. Florian öffnet die Listenversion.
- 8.1. Eine Liste mit Projekten und Veranstaltungen öffnet sich, die entsprechend der vorher eingegebenen Entfernung von der Route entfernt sind.
- 9.1. Die eingespeicherte Route bleibt unterhalb des Toggle Switches für ein späteres Anschauen erhalten.

5 AP 2: Konzept der Inhaltserstellung

KiezRadar soll Bürger:innen ermöglichen, sich leichter und gezielter an ihrer Stadt und deren Entwicklung zu beteiligen. Dafür sind Inhalte nötig, die Informationen über Beteiligungsmöglichkeiten enthalten. Diese müssen kategorisiert sein und einen Ortsbezug besitzen.

Mit diesen Voraussetzungen können sich Bürger:innen über bestimmte Ereignisse in ihrem Kiez oder auf ihrem Arbeitsweg informieren lassen und diejenigen für sich heraussuchen, die in ihrer derzeitigen Situation wichtig sind.

Grundsätzlich sind drei Arten der Informationsgewinnung dafür geeignet, die miteinander kombiniert werden können:

- 1 Nutzung öffentlich verfügbarer Quellen
- 2 Erstellung eigener Inhalte
- 3 Nutzung öffentlicher Quellen und Anreicherung der Inhalte

Im Projekt KiezRadar haben wir uns für die erste Variante entschieden, die Nutzung öffentlicher Quellen. Die beiden anderen Varianten wurden diskutiert und auch in den Workshops vorgestellt. Sie erhielten großen Zuspruch, waren aber aus praktischen Gründen im Prototyp nicht umsetzbar.

5.1 Nutzung öffentlich verfügbarer Quellen

Die Inhalte kommen in dieser Variante aus öffentlich verfügbaren Quellen. Das können Informationen der Stadt sein, von Behörden, anderen Beteiligungsplattformen oder von Bürgerinitiativen oder dem Quartiersmanagement.

In Berlin stehen sehr viele öffentliche Quellen zur Verfügung. Es sind so viele, dass eine Auswahl und Nutzung für interessierte Bürger:innen aufgrund des Umfangs schwierig ist.

Die für KiezRadar genutzten Inhalte werden aus den verfügbaren Daten extrahiert und leicht aufbereitet, insbesondere um Kategorien und, wenn nötig, um den Ortsbezug ergänzt.

Die Prüfung der Qualität der Inhalte obliegt in diesem Fall den bereitstellenden Quellen. Die bereitgestellten Inhalte werden unverfälscht in KiezRadar genutzt.

Das hat den Vorteil, dass Inhalte verschiedener Quellen schnell einheitlich genutzt werden können, ohne dass die Bürger:innen jede Quelle einzeln ansteuern müssen. Außerdem wird gezeigt, dass eine übergreifende Nutzung verschiedener Datenquellen auch die Verwaltungen entlastet, die nicht mehrere Systeme mit Daten »füttern« müssen.

Der Nachteil bei diesem Ansatz liegt darin, dass Verknüpfungen zwischen den Daten nicht gefunden werden, solange sie nicht in den Daten selbst abgelegt sind. Ob z. B. ein Ereignis einer Quelle das gleiche Ereignis beschreibt wie das einer anderen Quelle, ist nicht herauszufinden.

Das Problem könnte maschinell behoben werden, dies ist jedoch für einen Prototypen zu aufwendig.

5.2 Erstellung eigener Inhalte

Die Erstellung eigener Inhalte wurde im Rahmen der Anforderungsermittlung als ein großer Wunsch für KiezRadar festgehalten und konnte im Rahmen des Prototyps leider nicht geleistet werden. Der Aufwand eines Redaktionssystem, also ein entsprechendes Administrationsportal mit Zugriffssteuerung, Rollen und Rechten sowie der Abgleich mit vorhandenen Quellen, hätte den geplanten Rahmen deutlich überschritten.

Dies ist jedoch für die Akzeptanz von KiezRadar ein wichtiger Punkt, denn hier geht es darum, lokale Stakeholder einfach an die Informationsbeschaffung anzubinden. Das kann z. B. ein Architekturbüro sein, das eine lokale Informationsveranstaltung über ein Projekt anbietet oder ein Café, das einen Bürger:innenaustausch organisiert und nicht die Ressourcen für ein eigenes Redaktionssystem oder einen Onlinekalender besitzt.

Für die Weiterentwicklung von KiezRadar wäre dieser Punkt sehr wichtig, um die lokale Akzeptanz der App zu steigern.

5.3 Nutzung öffentlicher Quellen und Anreicherung der Inhalte

Hier ist vor allem eine redaktionelle Begleitung vorhandener Informationen gemeint, die ausgiebig im Bürger:innen-Workshop diskutiert und für wichtig befunden wurde. Mit entsprechenden Mitteln ist hier eine Aufbereitung von Projekten möglich, die dem Rahmen der einfachen Informationsbereitstellung überschreitet.

Das Spektrum der Anreicherung kann von Ankündigungen über Aufbereitung von Ergebnissen bis zur Nachverfolgung von Projekten reichen.

Auch diese Variante war im Rahmen des Prototyps nicht zu leisten, insbesondere, weil neben der technischen Struktur noch ein Redaktionsteam nötig ist, um Projekte zu betreuen. Im Verwaltungsworkshop sollte geklärt werden, inwieweit sich Verwaltungen und deren Pressestellen an so einer Idee beteiligen würden, dazu kam es jedoch aufgrund der ungeplanten Virtualität des Workshops nicht mehr.

Daher ist die Idee einer redaktionellen Anreicherung der Inhalte dazu geeignet, die Akzeptanz von KiezRadar zu erhöhen und sollte bei einer Fortführung bedacht werden, sie ist jedoch auch die aufwendigste Form.

5.4 Inhaltserstellung im Prototyp

Im Prototyp wurden Inhalte der Ratsinformationssysteme der Bezirke, Projekte und Pläne von mein.berlin.de sowie der Veranstaltungskalender des Quartiersmanagements Auguste-Viktoria-Allee genutzt.

Die Daten wurden über ihre jeweiligen Schnittstellen abgerufen und der Teil extrahiert, der für KiezRadar nötig ist.

Dabei wurde ein Informationsmodell der eingehenden Daten erarbeitet und dieses mit dem Informationsmodell der KiezRadar-Daten abgeglichen. Somit ist eine gezielte Umwandlung der Daten in die KiezRadar-Form reproduzierbar und automatisierbar möglich.

Danach wurde versucht, bei Daten ohne Geolokalisation diese aus der Beschreibung der Ereignisse abzuleiten.

Zusätzlich fand eine Kategorisierung der Daten anhand einer Ontologie⁶ statt. Diese Ontologie ist derzeit noch eine einfache Begriffszuordnung. Der Mechanismus ist jedoch auf unterschiedliche Repräsentationen ausgelegt, so dass auch andere Ontologierepräsentationen verwendet werden können, wie Tupel, semantische Netze oder ähnliches.

Diese aufbereiteten Daten sind die »KiezRadar-Daten«, die für die App verwendet werden.

Die verwendete Kategorienontologie ist:

```
{
  "Antidiskriminierung": [
    "ANT"
  ],
  "Arbeit & Wirtschaft": [
    "WOR"
  ],
  "Bauen & Wohnen": [
    "BUI"
  ],
  "Bildung & Forschung": [
    "EDU",
    "Bildung"
  ],
  "Kinder, Jugend & Familie": [
    "CHI",
    "Familie",
    "Jugendliche",
    "Kinder",
    "Senioren"
  ],
  "Finanzen": [
    "FIN"
  ],
  "Gesundheit & Sport": [
    "HEA",
    "Gesundheit"
  ],
  "Integration": [
    "INT"
  ],
  "Kultur & Freizeit": [
    "CUL",
    "Kultur"
  ],
  "Nachbarschaft & Teilhabe": [
    "NEI",
    "Info-Veranstaltungen",
    "Nachbarschaft"
  ],
  "Quartiersmanagement": [
    "Aktuell",
    "Info-Veranstaltungen",
    "Quartiersmanagement",
    "Regelmäßige Termine"
  ],
  "Stadtentwicklung": [
    "URB"
  ],
  "Umwelt & Grünflächen": [
```

⁶ grob gesagt: Begriffe und ihre Beziehungen zu anderen Begriffen, anhand derer eine Einordnung vorgenommen werden kann

```
    "ENV",  
    "Umwelt"  
  ],  
  "Verkehr": [  
    "TRA"  
  ]  
}
```

6 AP 3: Technisches Konzept, technische Architektur

Wie im vorigen Kapitel erläutert, werden die Inhalte für KiezRadar aus bestehenden Quellen genommen und für KiezRadar aufbereitet. Die so aufbereiteten Daten werden über das Backend der KiezRadar-App oder anderen Anwendungen zur Verfügung gestellt. Abbildung 18 zeigt dieses Konzept.

Abbildung 18: Technisches Konzept von KiezRadar.



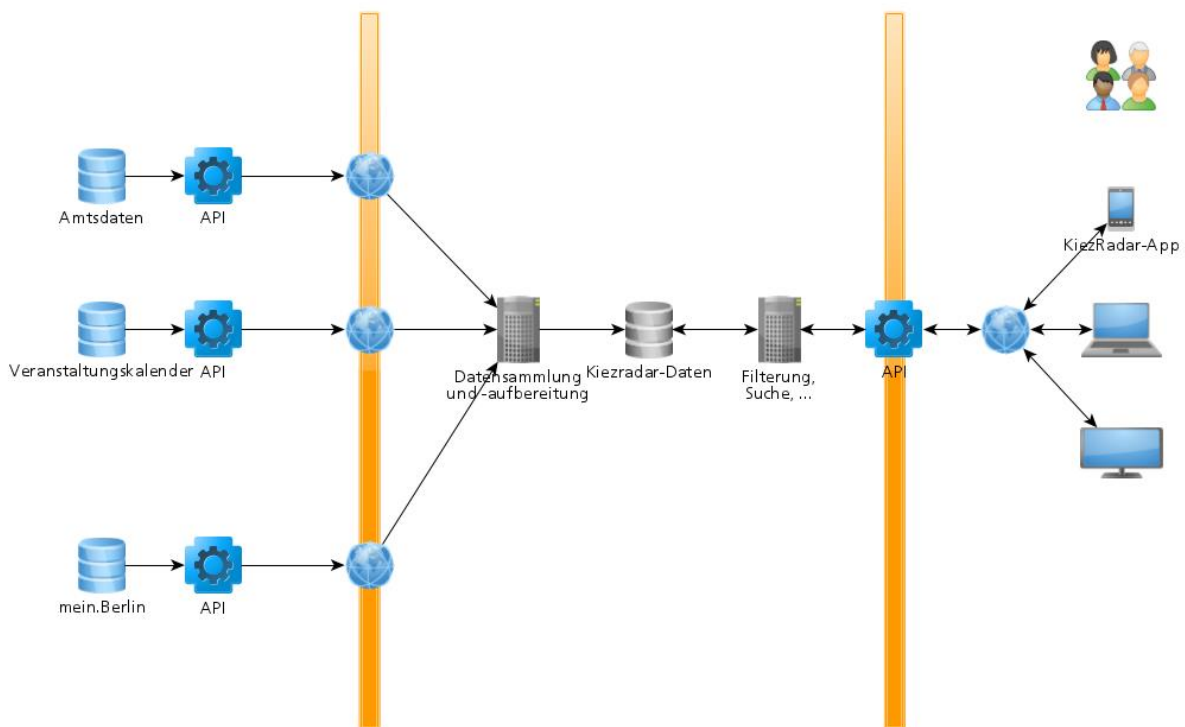
Technisch erfolgt die Anbindung über die Schnittstellen der Quellen. Eine zusätzliche Komponente ist dafür nicht nötig.

Die importierten Daten werden in das kiezradareigene Datenformat überführt und, wenn nötig, dafür aufbereitet. Die Aufbereitung enthält insbesondere die Kategorisierung und Geolokalisierung der Daten. Die so gewonnenen Daten werden in einer Datenbank gespeichert. Dafür sind sogenannte Adaptionen zuständig.

Die Inhalte der Datenbank werden über eine REST-Schnittstelle öffentlich zur Verfügung gestellt. Die KiezRadar-App greift auf diese Schnittstelle zu, sie ist jedoch auch für andere Anwendungen offen.

Abbildung 19 zeigt die zugehörige technische Grobarchitektur.

Abbildung 19: Technische Grobarchitektur von KiezRadar.



6.1 Architekturprinzipien

Der Architektur liegen mehrere Prinzipien zugrunde:

- Entkopplung der Quelldaten von den KiezRadar-Daten
- Eigene Datenhaltung und eigenes Datenformat für KiezRadar
- Einheitliche KiezRadar-API gegenüber der KiezRadar-App und anderen Anwendern

6.1.1 Entkopplung der Quelldaten von den KiezRadar-Daten

Die Quelldaten liegen in unterschiedlichen Systemen und unterschiedlichen Formen, Formaten oder Granularität vor. Gegenüber den KiezRadar-Nutzer:innen sollen die Daten jedoch einheitlich vorliegen. Daher ist eine Datenaufbereitung für KiezRadar unerlässlich.

Die Datenaufbereitung geschieht, indem die Daten der unterschiedlichen Quellsysteme ausgelesen und vereinheitlicht werden. Dafür wurde ein eigenes Datenformat geschaffen, in das die Daten überführt werden. Somit können redundante oder widersprüchliche Daten erkannt und aufgelöst werden. Es ist auch möglich, fehlende Daten aus unterschiedlichen Quellen durch Kombination zu ergänzen. Das Datenformat ist die Grundlage der nach außen sichtbaren API und wird weiter unten beschrieben.

Damit das funktioniert, sind zum einen wohldefinierte Schnittstellen an den Quellsystemen notwendig. Dies betrifft nicht nur die technische Ausgestaltung der Schnittstellen, sondern auch deren Semantik. Derzeit ist das vor allem die Kategorisierung der verschiedenen Ereignisse in »Bauen & Wohnen«, »Bildung & Sport« etc. Wenn nötig, wird das anhand von Ontologien durchgeführt, die eine konkrete Zuordnung ermöglichen.

Jede Datenquelle wird mit einem speziellen Adaptor abgefragt, der zum einen die Abfrage übernimmt, zum anderen aber auch die Datenaufbereitung. Nach dieser Aufbereitung sind die Daten zur Geolokalisierung bereit.

In diesem Schritt werden alle Ereignisse daraufhin geprüft, ob bereits Ortsangaben in Form von Geokoordinaten vorhanden sind. Ist dies nicht der Fall, wird versucht, aus dem angegebenen Ort die Koordinaten zu ermitteln.

Jetzt sind die Daten so aufbereitet, dass sie in die interne Datenhaltung überführt werden können.

Die Entkopplung der Daten bietet Vor- und Nachteile, die im Folgenden aufgezeigt werden:

Vorteilhaft ist, dass die Last auf den Quellsystemen verringert wird, da nicht für jede KiezRadar-Abfrage eine Anfrage der Quellen geschehen muss. Ein Live-Zugriff auf die Quellsysteme für jede Abfrage ist nicht nur aus Lastgründen unrealistisch, auch die Aufbereitung muss vorgenommen werden, sie benötigt Zeit, die die KiezRadar-API verzögern würde. Zusätzlich bekommt die KiezRadar-Auslieferungseite so genügend Möglichkeiten, ein vernünftiges Cachen der Daten zu ermöglichen.

Zusammengefasst sind die Vorteile:

- weniger Last auf den Quellen
- Möglichkeit zur Aufbereitung, Verknüpfung und Anreicherung der Daten
- Vereinheitlichung der Daten
- eigene Auslieferung und damit Entkopplung von der Leistungsfähigkeit und Zuverlässigkeit der Quellsysteme

Der Ansatz birgt zwei hauptsächliche Nachteile: erstens sind die Daten nur so aktuell wie bei der letzten Abfrage der Quellen. Das heißt, Änderungen der Daten werden erst nach einem erneuten Abgleich in KiezRadar sichtbar. Zweitens muss für die Datenhaltung ein eigener Datenbankserver vorgehalten werden.

Beide Nachteile werden bei KiezRadar durch die beschriebenen Vorteile aufgewogen. Außerdem ist die Aktualität der Daten in den meisten Fällen nicht kritisch, da die Ereignisse eher in Wochen und Monaten als in Stunden geplant werden. Auch der Datenbankserver ist problemlos umzusetzen, da für die Abfrage und Bereitstellung der KiezRadar-API sowieso ein eigener Server nötig ist.

Zusammengefasst sind die Nachteile:

- Aktualität der Daten nur zum Abfragezeitpunkt
- eigene Datenbank zur Datenhaltung nötig

6.1.2 Eigene Datenhaltung und eigenes Datenformat für KiezRadar

Eine eigene Datenhaltung sowie ein internes Datenformat sind für KiezRadar unerlässlich.

Zunächst können die Adaptoren, die den Datenimport und die Datenaufbereitung übernehmen, zeitunkritisch und lastunabhängig ihre Daten in die KiezRadar-Infrastruktur liefern. Etwaige Lastspitzen auf beiden Seiten (Datenlieferung und -abfrage) können jeweils durch KiezRadar-interne Mechanismen abgefangen und behandelt werden. Somit werden die Quellsysteme gleichmäßig und seltener belastet als dies durch direkten Zugriff nötig

wäre. Außerdem kann die Datenpflege »in Ruhe« vorgenommen werden, anfänglich muss z. B. getestet werden, ob die Datenpflege unter Umständen manuell vorgenommen werden muss, bis alle Quelldaten harmonisiert und vollständig vorliegen. Auf der Abfrageseite kann die Datenbank entsprechend skaliert werden, dass Lastspitzen durch entsprechende Mechanismen wie Caching oder Datenbankreplikation abgefangen werden können.

Den Adaptoren und der KiezRadar-API wird eine einheitliche Zugriffsschicht geschaffen, gegen die programmiert werden kann. Die Datenquellen und -senken sind damit entkoppelt. Das sorgt für stabilere und einfachere Schnittstellen auf beiden Seiten.

Weiterhin kann das Datenmodell auf die KiezRadar-Daten reduziert werden. Die Quellsysteme liefern deutlich umfangreichere und vielfältigere Daten als für KiezRadar benötigt werden. Die Datenreduktion geschieht in den Adaptoren. Die Datenbank und die KiezRadar-API werden davon befreit. Das senkt die Gesamtkomplexität der Anwendung und eröffnet mehr Skalierungsmöglichkeiten, falls diese benötigt werden.

Das Datenmodell wird für die KiezRadar-API ebenfalls benötigt, so dass hier eine gleichzeitige Entwicklung möglich ist.

Wie bereits beschrieben, hat die eigene Datenhaltung den Nachteil, dass eine eigene Datenbank mit einem entsprechenden Datenschema notwendig ist. Das bedeutet Implementierungs- und Wartungsaufwand. Außerdem ist ein Server für die Datenbank nötig.

Beides ist für KiezRadar unkritisch, da für die API sowieso ein Server benötigt wird und auch das Datenbankschema über die Syntax der API definiert ist.

6.1.3 Einheitliche KiezRadar-API

Gegenüber der KiezRadar-App und anderen Anwender:innen stellt KiezRadar eine eigene REST-API zur Verfügung.

Damit wird der Zugriff auf die Daten von den Quellen entkoppelt, nicht nur physisch, sondern auch von der Syntax und Semantik der Daten. Die KiezRadar-API stellt die notwendigen Daten für die Anwendung zur Verfügung und bietet diese in einem einheitlichen Format an, egal von welcher Quelle die Daten stammen.

Damit ist die API gegenüber dem Direktzugriff einfacher, schneller und präziser. Außerdem können die Anwendungsszenarios der API konkreter dargestellt werden, die Untersuchung auf Datenschutz oder Datenmissbrauchsszenarios ist einfacher möglich und direkt lösbar. Insbesondere im Hinblick auf eine perspektivische Nutzung der Daten für eine OpenAPI-Schnittstelle ist das sehr praktikabel.

Die Schnittstelle wurde zunächst an OParl angelehnt und dann auf die Bedürfnisse von KiezRadar zugeschnitten.

6.2 KiezRadar-API

Die KiezRadar-API⁷ bietet Anwendungen einen einheitlichen, offenen Zugriff auf die Daten. Sie ist in der technischen Architektur (Abbildung 18) direkt eingezeichnet, im

⁷ »application programming interface« = Anwendungsschnittstelle

technischen Konzept (Abbildung 19) wird sie durch den rechten, ausgehenden Pfeil von KiezRadar symbolisiert.

Die API kann über folgende URL angesprochen werden:

<https://kiezradar.fokus.fraunhofer.de/api/>

Der direkte Aufruf liefert die technische Beschreibung der API zurück, die das Datenmodell sowie alle gültigen Aktionen der API enthält.

Die API ist REST⁸-basiert, das heißt, sie kann als normale REST-API mit entsprechenden POST-Methoden angesprochen werden.

Die Modellierung der API erfolgt mit Hilfe einer OpenAPI-Spezifikation, das heißt, mit Hilfe eines offenen Standards zur Beschreibung von REST-konformen Schnittstellen. So wird sichergestellt, dass die API standardisiert angesprochen werden kann. Das erleichtert den Zugriff auf die Schnittstelle.

Die Datenmodellierung der Schnittstelle ist lose an OParl angelehnt, dort, wo das für die Modellierung nötig ist. OParl ist ein Standard der Open Knowledge Foundation für den Datenzugriff auf parlamentarische Informationssysteme.

Im Kontext von KiezRadar ist die Modellierung von Tagesordnungspunkten und Orten interessant. Ein großer Teil des OParl-Standards, der den Aufbau von föderalen Strukturen widerspiegelt, wurde nicht verwendet. Für die Modelle von KiezRadar sind zusätzliche Informationen nötig, die entsprechend in der OpenAPI-Spezifikation hinterlegt wurden.

OParl schreibt bestimmte Inhalte in seiner Spezifikation richtigerweise als zwingend notwendig vor. Die Umsetzung hätte für KiezRadar jedoch bedeutet, viele Objekte der Form halber mitzuführen und den Zugriff auf die KiezRadar-Objekte zu verkomplizieren.

Daher haben wir nach reiflicher Abwägung entschieden, dass die KiezRadar-Schnittstelle nicht OParl-konform wird und so die Freiheiten genutzt, die API passgenau auf das Anwendungsszenario von KiezRadar anzupassen.

Neben der technischen Argumentation ist das auch inhaltlich vertretbar, da KiezRadar zwar Daten auch aus Ratsinformationssystemen nutzt, jedoch nur einen Auszug davon. Die kompletten Inhalte von OParl werden durch KiezRadar nicht geliefert.

Perspektivisch ist es möglich, einen API-Layer zu konzipieren, der einen zusätzlichen OParl-konformen Zugriff auf die Daten ermöglicht. Dies wurde jedoch weder konzipiert noch umgesetzt, da das Anwendungsszenario für diesen Anwendungsfall noch unklar ist.

Die API und deren Dokumentation kann über die oben angegebene URL abgerufen werden, sie ist auch als Open Source in gitlab veröffentlicht. Somit haben Anwender und Entwickler jederzeit die Möglichkeit, sich mit der API vertraut zu machen und sie nach eigenem Ermessen auch anzupassen.

⁸ »Representational State Transfer« - Paradigma für Webservices

6.3 Datenaufbereitung

Die Daten, die von den verschiedenen Systemen abgerufen werden, sind sehr unterschiedlich. Das betrifft die Quantität der Informationen, die zu einem Ereignis geliefert werden aber auch die Art der Informationen, die geliefert werden.

Derzeit werden von KiezRadar folgende Datenquellen eingebunden:

- Veranstaltungskalender der Bezirksverordnetenversammlungen der Bezirke (Allris-XML-Schnittstelle)
- mein.berlin.de-Pläne und -Projekte (mein.Berlin.de-json-Schnittstelle)
- Kalender des Quartiersmanagement Auguste-Viktoria-Allee (ical-Schnittstelle)

Die erste Aufgabe ist der Import der Daten, dieser ist durch die Standards der Schnittstellen einfach möglich.

Die zweite Aufgabe ist die Aufbereitung. Für KiezRadar sind neben dem Titel und einer Beschreibung eines Ereignisses, dessen Zeit, Ort und Kategorisierung wichtig. Alle Datenquellen liefern einen Titel und ein Datum, der Rest der Daten fehlt oder wird in unterschiedlichen Datenformaten geliefert.

Daher ist zunächst eine technische Aufbereitung der Daten notwendig, so dass alle Daten in einem einheitlichen Format vorliegen. Ein Beispiel sind Datum- und Zeitangaben, die als Datumstext, Zeittext oder als Kombination der beiden Texte in einem Zeitstempel geliefert werden. Hier stehen nach der Aufbereitung ein Datum und eine Zeit zur Verfügung.

Daran schließt sich die inhaltliche Aufbereitung an. Hier wird anhand von Ontologien, derzeit anhand von Kategorietupeln, eine Kategorisierung der Daten vorgenommen (siehe Abschnitt 5.4).

Außerdem werden diejenigen Texte herausgezogen, anhand derer bei fehlenden Geodaten eine Geolokalisierung vorgenommen werden kann.

Auf diesem Weg werden alle Daten in ein datenquellenunabhängiges Zwischenformat gebracht, das dem Geolokalisator zugeführt wird. Dieser versucht, anhand der Ortstexte mittels der Openstreetmap-Lokalisierung die Geokoordinaten der Orte herauszufinden und damit die Daten anzureichern. Die so angereicherten Daten werden in einem zweiten Zwischenformat abgelegt.

Dieses Zwischenformat wird nun genutzt, um die Datenbankinhalte zu erstellen, also die Datenbank mit den Daten im KiezRadar-Format zu füllen.

6.4 Frontend

Als Frontend wurde in KiezRadar eine native Android-App konzipiert und umgesetzt.

Die App nutzt die API von KiezRadar, um die Daten zu importieren. Dann werden die Daten verarbeitet und den Nutzer:innen angezeigt.

Die App bietet die Möglichkeit, die Daten zu filtern oder die Ergebnisse zu verfeinern. Diese Einschränkung der Suche geschieht in der App auf dem Handy, basierend darauf wird die API erneut aufgerufen, die die gefilterten Ergebnisse zurückgibt.

Das heißt, die App selbst soll keine Filterung oder Suchen ausführen, sondern dafür die Funktionen der KiezRadar-API nutzen. Damit benötigt die App keine Speicherung der KiezRadar-Daten.

Die einzige vorgesehene Speicherung in der App ist die von Voreinstellungen wie Standort, Favoriten oder beliebte Filter. Diese Daten werden lokal im internen Speicher der App gehalten und nicht zentral auf dem Server gespeichert.

Diese Architekturentscheidung ermöglicht es, den Server zunächst ohne Rückkanal, also ohne Datenspeicherung von Nutzerdaten, zu konzipieren. Das erleichtert die datenschutzrechtliche Einschätzung der Datenverarbeitung, denn Nutzerdaten werden ausschließlich in der Hoheit der Nutzer:innen gespeichert, nicht zentral.

Außerdem wird die App-Entwicklung von der Speicherung der KiezRadar-Daten befreit, so dass Caching mit seinen Synchronisationsproblemen nicht betrachtet werden muss.

Nachteilig ist an diesem Ansatz, dass die App für die Nutzung online sein muss. Eine Offline-Nutzung ist so nicht möglich.

Für eine Prototypentwicklung wurde so eine pragmatische Entscheidung getroffen, die viele zeitaufwendige Elemente der Implementierung vermeidet, die jedoch für eine voll funktionsfähige App bedacht werden müssen.

7 AP 4: Entwicklung

In diesem Kapitel wird kurz die Entwicklung des KiezRadar-Prototyps beschrieben, das Kapitel ist jedoch keine technische Beschreibung der implementierten Artefakte. Vielmehr wird erläutert, wie auf Basis der technischen Architektur eine funktionsfähige Lösung gefunden wurde und welche Entscheidungen dabei aus welchen Gründen gefällt wurden.

Die Entwicklung von KiezRadar erfolgt in drei unterschiedlichen technischen Bereichen:

- Datenimport und -aufbereitung
- Backend (Datenspeicherung und Schnittstelle)
- Frontend (App)

Dabei werden die Bereiche »Datenimport und -aufbereitung« im gleichen Repository gehalten, also gemeinsam entwickelt, da hier sehr viele Überschneidungen vorhanden sind und eine Trennung keinen Vorteil im Prototypen ergeben hätte.

7.1 Plattformen und Docker

Allgemein wurde versucht, die Entwicklung so plattformunabhängig wie möglich zu halten, was die Entwicklungswerkzeuge angeht. Daher wurde, wo möglich, auf Docker als zugrundeliegende Plattform zugegriffen, da Docker sowohl für Windows, Linux als auch für macOS verfügbar ist.

Für die App-Entwicklung wurde Android-Studio eingesetzt, das ebenfalls für diese Plattformen zur Verfügung steht.

Beides hat sich im Projekt bewährt, da so keine Verluste bei einem Wechsel der Plattform entstehen und die Entwicklung sehr schnell nachvollzogen werden kann.

Als Zielplattformen wurden für das Backend Ubuntu mit Docker und für die App Android gewählt. Dies ist für den Prototypen ausreichend und ermöglicht eine agile Herangehensweise an die Entwicklung.

Bei der Wahl der Zielplattformen wurde ebenfalls auf bewährte Mittel zurückgegriffen: Ubuntu mit Docker als Server ist eine bewährte Wahl. Für die App hätte auch eine Web-App gewählt werden können, die Verbreitung von Android und hauseigenes Know-How gaben jedoch den Ausschlag für Android, da insbesondere bei solch kurzen, schnellen Projekten keine lange Einarbeitungszeit im Weg stehen sollte.

7.2 Datenimport und -aufbereitung

Für den Datenimport wurden Python-Skripte genutzt.

Python als Scriptsprache ist ein gutes Mittel, um schnell Dinge ausprobieren zu können und auf Veränderungen reagieren zu können. Außerdem ist keine große Infrastruktur nötig, um Python-Skripte laufen zu lassen.

Python bietet über seine Bibliotheken zudem alle Standards, die für den Zugriff auf Web-Schnittstellen benötigt werden.

Folgende Datenquellen wurden genutzt:

- Veranstaltungskalender der Bezirksverordnetenversammlungen der Bezirke (Allris-XML-Schnittstelle)
- mein.berlin.de-Pläne und -Projekte (mein.berlin.de json-Schnittstelle)
- Kalender des Quartiersmanagement Auguste-Viktoria-Allee (ical-Schnittstelle)

Die zu verarbeitenden Daten sind dementsprechend XML, JSON und iCal. Insgesamt sind ca. 7.000 Datensätze zu verarbeiten.

Die Daten werden direkt von der Schnittstelle importiert. Die Veranstaltungskalender enthalten die Ereignisse selbst als Tagesordnungspunkte. Wenn es Details zu einzelnen Ereignissen gibt, müssen diese explizit abgefragt und verarbeitet werden. Bei mein.berlin.de wird für jedes Ereignis ein Datensatz übermittelt. Ein iCal-Kalender enthält alle Ereignisse in einem Datensatz, der in einzelne Ereignisse zerlegt werden muss.

Dann werden die für KiezRadar relevanten Daten herausgezogen und alle Daten in ein intermediäres JSON-Format überführt, das neben den Originaldaten die neuen KiezRadar-Daten enthält.

Basierend auf diesen Metadaten wird die Kategorisierung vorgenommen.

Die Ereignisse werden in folgende Kategorien eingeordnet:

- Antidiskriminierung
- Arbeit & Wirtschaft
- Bauen & Wohnen
- Bildung & Forschung
- Kinder, Jugend & Familie
- Finanzen
- Gesundheit & Sport
- Integration
- Kultur & Freizeit
- Nachbarschaft & Teilhabe
- Quartiersmanagement
- Stadtentwicklung
- Umwelt & Grünflächen
- Verkehr

Die so aufbereiteten Daten werden nun der Geolokalisierung unterzogen. Dabei wird versucht, den übermittelten Adressen eine Geokoordinate zuzuweisen. Dafür wird der Nominatim-Mechanismus von Openstreetmap genutzt.

Dieser Mechanismus kann selbst ausprobiert werden:

<https://nominatim.openstreetmap.org/>

Dabei waren zwei Probleme zu überwinden. Erstens darf der Server für privaten Zwecke benutzt werden, jedoch ist es nicht erlaubt und kein adäquates Verhalten, den Server für derartige Massenfragen zu benutzen, die bei KiezRadar anfallen.

Daher muss ein Nominatim-Server lokal aufgesetzt werden. Dafür wurde ebenfalls Docker genutzt. Leider ist der Nominatim-Server der einzige Container, der zwingend einen Linux-Host benötigt.

Zum Initialisieren des Servers benötigt man eine Datei mit Ortsangaben von Berlin, die frei heruntergeladen werden kann, z. B. von [bbbike.de](https://download.bbbike.org/osm/bbbike/Berlin/)⁹

Zweitens sind die Ortsangaben in den Quellen nicht normiert und enthalten neben Straße und Ort oft auch ergänzende Elemente, wie Hausnamen oder Adresszusätze. Das verhindert oft die erfolgreiche Lokalisierung.

Im Projekt wird der Text in diesem Fall nach und nach aufgeteilt, im Bestreben, einen lokalisierbaren Teil zu finden. Dieser Ansatz findet einige Treffer, ist jedoch auch sehr fehleranfällig. Hier muss die Lokalisierung über die Qualität der Eingangsdaten verbessert werden.

Die so aufbereiteten und mit Geokoordinaten versehenen Daten werden als JSON-Dateien für das Backend abgelegt.

7.3 Backend

Das Backend wurde mit einem API-First-Ansatz konzipiert. Das heißt, die API dient als Ausgangspunkt für die Entwicklung und bestimmt die Entwicklungsartefakte.

Als API-Beschreibung wurde eine OpenAPI-Spezifikation gewählt, OpenAPI ist der derzeitige de-facto-Standard für die Beschreibung offener Schnittstellen.

Als Datenbank wurde eine PostgreSQL-Datenbank gewählt, da sie zum einen eine bewährte Datenbank für Serveranwendungen ist und zum anderen mit PostGIS eine Erweiterung für die Speicherung von Geokoordinaten zur Verfügung steht.

Im ersten Versuch war geplant NodeJS als Server einzusetzen. Dieser Ansatz hilft, die API von der Datenbank zu entkoppeln und unterschiedliche Anfragen zu ermöglichen. Der Ansatz bedeutet jedoch auch, bei jeder Änderung der API auch den Servercode anpassen zu müssen.

Daher wurde von NodeJS zu PostgREST für die Auslieferung der Daten übergegangen. Bei diesem Ansatz wird die zur Verfügung gestellte API von PostgREST aus der Datenbank abgeleitet. Das hat den Vorteil, dass bei einer Änderung der Datenbank automatisch die API aktualisiert wird. Um diesen Vorteil auszunutzen, wird das Datenbankschema aus der OpenAPI-Spezifikation generiert.

Dieser Ansatz hat zwei Nachteile. Erstens weicht die von PostgREST generierte API von der in OpenAPI spezifizierten API ab. Zweitens wird eine Versionierung der API erschwert.

Nach einer Abwägung dieser Nachteile gegen die Vorteile der Generierung fiel die Entscheidung auf den API-First-Ansatz, da die Änderung der API im Projektverlauf häufiger zu erwarten war und der Aufwand für nachfolgende Anpassungen möglichst gering ausfallen sollte.

Im Produktivbetrieb müssen diese Probleme adressiert und gelöst werden, für eine Prototypentwicklung war der Ansatz sehr hilfreich.

Es ergibt sich also folgender Ablauf für die Generierung des Servers:

⁹ <https://download.bbbike.org/osm/bbbike/Berlin/>

- 1 Erzeugung des Datenbankschemas aus der OpenAPI-Spezifikation
- 2 Konvertierung der Daten in das Datenbankschema
- 3 Füllen der Datenbank mit den Daten
- 4 Start des PostgREST-Servers

Für die Erzeugung des Datenbankschemas aus der OpenAPI-Spezifikation wurden Werkzeuge der OpenAPI-Generator-Sammlung verwendet. Leider gab es keinen direkten Konverter OpenAPI zu PostgreSQL, so dass ein anderer Konverter verwendet und dessen Ergebnisse mit Python und FMPP in Form gebracht wurden. Das ist aufwendig und langfristig fehleranfällig, wenn der verwendete Konverter weiterentwickelt werden sollte. Wenn der Ansatz weiterverfolgt werden soll, ist ein eigener Generator zu erstellen, der direkt PostgreSQL-Code erzeugt.

Die von der Datenaufbereitung gelieferten JSON-Dateien werden mit Hilfe von FMPP in SQL-Befehle umgewandelt.

Aus diesen Daten wird eine SQL-Datei erzeugt, die die Datenbank anlegt und mit Daten füllt. Mit Hilfe dieser Datei wird die PostgreSQL-Datenbank gefüllt.

All diese Schritte werden über Docker ausgeführt. Für die Datenbank wird ein eigenes Docker-Image erzeugt.

Dieses Docker-Image wird über Docker-Compose mit dem PostgREST-Server verbunden, der nun die Daten ausliefern kann.

Die einzelnen Schritte sind in automatisierten Skripten abgelegt, müssen jedoch von Hand nacheinander aufgerufen werden. Das ist auch sinnvoll, da während der Entwicklung verschiedene Probleme mit Daten, Datenquellen etc. auftraten, die gelöst werden mussten. Jede Anpassung sollte so wenig Einfluss auf die anderen Schritte wie möglich haben, so dass eine Skriptmodularisierung unumgänglich war.

Derzeit sind die Datenquellen leider noch nicht stabil genug, um einen vollautomatisierten Ablauf von Datenimport bis Auslieferung zu garantieren, daher muss die Datenaktualisierung derzeit noch von Hand vorgenommen werden. Im Produktivbetrieb muss dieser Bereich daher noch gehärtet werden, damit die Datenaktualisierung in einem Cron-Job ablaufen kann.

7.4 Frontend

Das Frontend von KiezRadar ist der App-Prototyp.

7.4.1 Design und Funktionskonzept

Auf Basis der Anforderungen wurden Mockups für sieben wesentliche Funktionsbereiche erstellt:

- Entdecken
- Kartenansicht
- Favoriten
- Suche
- Einstellungen
- Filter
- Benachrichtigungen

Mockups bieten eine gute Möglichkeit, um verschiedene Designideen zu erproben und zu diskutieren. Zudem kommen bei der Erstellung viele Fragen zu Präferenzen oder Bedürfnissen der Nutzer:innen auf, die dadurch frühzeitig geklärt werden können. Bei den Designentwürfen wurde darauf geachtet, eine Vielzahl der erhobenen Anforderungen aus dem Bürger:innen-Workshop zu berücksichtigen. Zudem wurden die im UX-Konzept (siehe Kapitel 2) erarbeiteten Prinzipien und Vorgaben bei der Erstellung der Mockups berücksichtigt.

Die Mockups wurden sukzessive angepasst und verfeinert. Zudem dienten die grafischen Designentwürfe als Grundlage für die Erstellung von HTML-basierten Mockups.

Folgende Schritte wurden in der Entwicklung des Designs und Funktionskonzepts durchgeführt:

- Entwicklung eines Klickprototypen der App
- Erweiterung des Klickprototypen um Datensammlungs- und Verarbeitungsfunktionen
- Iterative interne Tests mit externen Datenquellen und Fachsystemen

Die konkrete Ausgestaltung, Rückmeldungen und UI-Überarbeitungen können Sie in den Kapiteln zu Arbeitspaket 1 nachlesen.

7.4.2 Android App Implementierung

Auf Basis der Mockups und eines ersten Funktionskonzepts wurde ein erster Prototyp entwickelt. Neben dem grundlegenden Design der App, lag der Fokus dabei vor allem auf der Umsetzung der entscheidenden Kernfunktionen: Abrufen der – mit Filtern eingegrenzten – Events von einer Schnittstelle, Anzeigen der Events auf einer Openstreetmap-Karte und Anzeigen der Events in einer Listenansicht.

Nach einer ersten internen Evaluierung, die auf Basis bestehender innovativer Implementierungen getroffen wurde, wurden zuerst Mockups und danach auch die Implementierung angepasst. Demnach war in der ersten Entwicklungsphase vor allem die Rückkopplung ins Team sehr wichtig, um Schwachstellen des Designs und der beschriebenen Funktionen aufzudecken und die App weiterzuentwickeln. Interne Funktionstests sind in dieser Entwicklungsphase unerlässlich und zeigen nicht nur Schwachstellen auf, sondern fördern auch neue Ideen für Design und Funktionen. Bereits zur Mitte dieser Phase wurde mit echten Datensätzen aus den beschriebenen externen Datenquellen gearbeitet, um das Nutzererlebnis so realitätsnah wie möglich zu testen.

In einem nächsten Schritt konnte durch die Durchführung von Nutzertests (siehe Kapitel 8) direktes Feedback potentieller Nutzer:innen in die Entwicklung einbezogen werden. Nach einer systematischen Auswertung wurden weitere Anforderungen aufgenommen, die dann teilweise in der Entwicklung des Prototyps umgesetzt wurden. Neue Vorschläge und Anforderungen aus den Nutzertests waren dabei hilfreich, Ideen auszutesten und anhand von direktem Feedback zu evaluieren. Aufgrund der priorisierten Umsetzung der Anforderungen, wurden zunächst spezielle Kernfunktionen der App bewertet. Das Eingeben und Nutzen von Filtereinstellungen ist hierbei die Funktionalität, die am meisten mit dem Feedback von Nutzer:innen weiterentwickelt wurde (siehe Abbildung 20). Auch Designentscheidungen bei der Anzeige von Events mithilfe von Markern auf der Kartenansicht wurde, durch die Evaluation durch Nutzer:innen optimiert (siehe Abbildung 21).

Abbildung 20: Filterfunktion, links vor der Umgestaltung anhand des Nutzer:innen-Feedbacks und rechts danach.

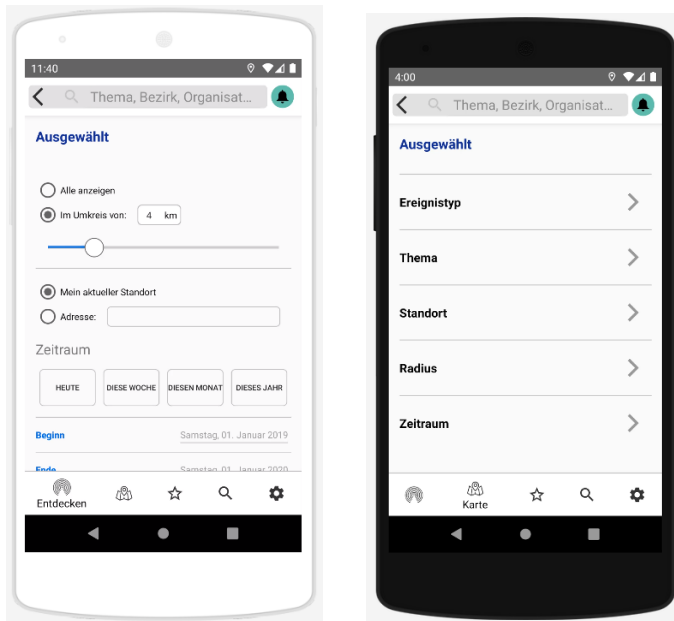
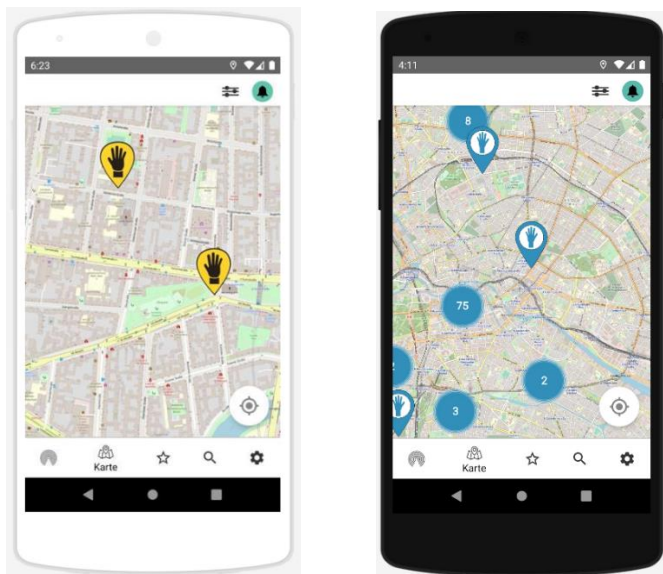


Abbildung 21: Präsentation der Events, links vor der Umgestaltung anhand des Nutzer:innen-Feedbacks und rechts geclustert.



Das Resultat der Entwicklung im Frontend ist ein sehr detailliertes Design in Form von Mockups, die durch den Prototyp greifbar gemacht werden. Bei der Implementierung wurden allerdings nicht alle Funktionen, die eine solche App entsprechend der Nutzeranforderungen umfassen könnte, umgesetzt, da dieses der zeitlichen Rahmen des Projekts nicht vorhergesehen hat und es den Rahmen eines Klickprototypen sprengen würde.

Im Prototyp umgesetzt wurden die Bereiche Entdecken-Funktion, Kartenansicht, Favoritenfunktion und Filterfunktion (siehe Abbildung 22). Außerdem umgesetzt wurde die Detailansicht der Ereignisse (siehe Abbildung 23)

Abbildung 22: Screenshots der im Prototypen umgesetzte Funktionsbereiche.

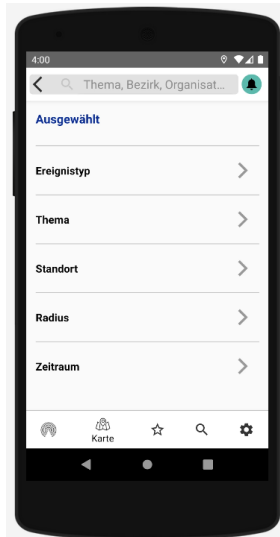
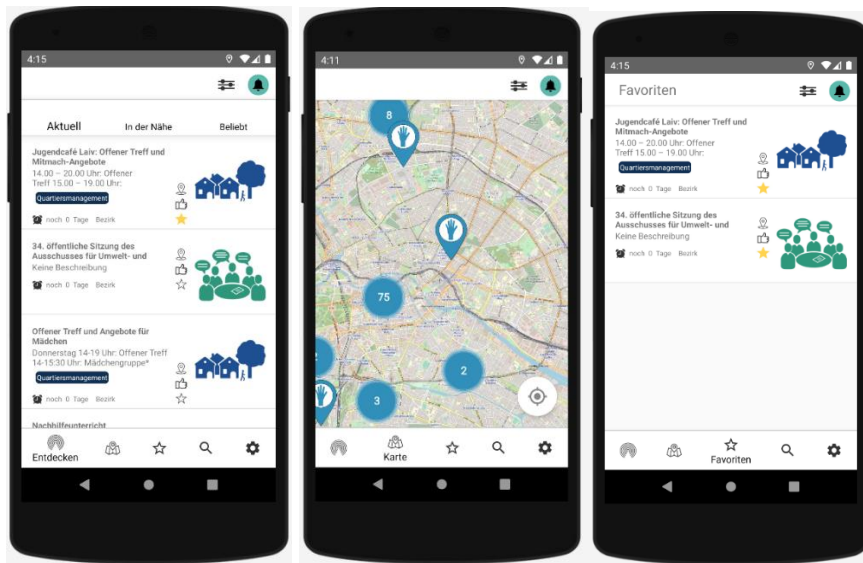
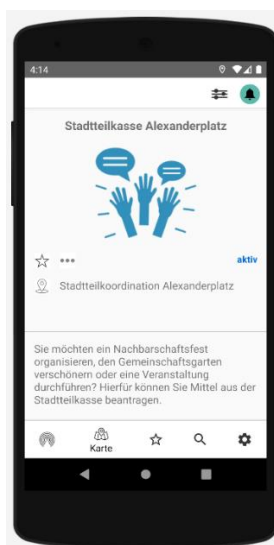


Abbildung 23: Screenshot der im Prototypen umgesetzten Detailansicht.



Besondere Herausforderungen bei der Umsetzung einer solchen App-Implementierung waren die asynchrone Ausführung bestimmter Programmteile, die Umsetzung eigener Designideen mit bestehenden Designelementen aus dem Android-Kosmos und verschlüsselte Schnittstellenkommunikation. Zudem ist speziell die von Android vorgegebene Lösung für Schnittstellenkommunikation nur für sehr neue Versionen von Android geeignet und schränkt damit die Abwärtskompatibilität zu alten Geräten extrem ein. Eine native App-Implementierung bietet durch diverse systemspezifische Eigenheiten weitere Herausforderungen.

7.4.3 Abweichungen vom Architekturkonzept

An zwei Stellen wurde vom Architekturkonzept abgewichen:

Das Konzept sieht vor, jegliche Datenfilterung per API durchzuführen. Diese kann jedoch die Filterung nach Kategorien noch nicht in Kombination mit anderen Filtern ausführen. Daher wird die Kategoriefilterung in der App vorgenommen.

Die zweite Abweichung betrifft die lokale Speicherung von Ereignisdaten. Um in Offlinephasen oder bei Verbindungsproblemen Events anzeigen zu können, werden die Ereignisse der letzten Abfrage der API auf dem Handy gespeichert.

8 AP 5: Test- und Pilotphase

8.1 Planung

Durch die Zusammenarbeit mit dem CityLAB Berlin war es möglich, die Evaluation des Pilotsystems größtenteils im Rahmen der Falling Walls – Berlin Science Week durchzuführen. Unter dem Titel »KiezRadar zum Testen« wurde die Teilnahme am Test des Pilotsystems im Programm der Berlin Science Week beworben.

Viele Personen zeigten Interesse an einem Test, wodurch eine diverse Gruppe von Testpersonen entstand. Insgesamt wurden zwölf Tests vom 9.11. bis 11.12.2020 durchgeführt.

Die Evaluation war ursprünglich als physisches Event mit Ein-Personen-Slots in den Räumlichkeiten des CityLABs geplant. Hierfür wurde ein Konzept erstellt und Materialien organisiert. Aufgrund der Erhöhung der Infektionszahlen mit COVID-19 in Berlin im Laufe des Oktobers musste die Evaluation im November jedoch digital durchgeführt werden.

8.2 Ablauf

Als Interface für die App wurde ein webbasierter Emulator genutzt. Mit einem webbasierten Emulator kann per Link auf ein mobiles Endgerät im Browser zugegriffen werden, auf welchem die KiezRadar-App ausgeführt wird. Für noch nicht implementierte Features wurden Mockups gezeigt.

Die Evaluation basiert auf dem Testen der KiezRadar-App durch Nutzer:innen unter Anwendung der Methode Lautes Denken und dem Stellen von Fragen. In der Methode des Lauten Denkens werden Nutzer:innen gebeten, eine spezifische Aufgabe in der Anwendung durchzuführen und dabei ihre Gedanken auszusprechen.

Der Test umfasste ein anfängliches freies Explorieren der App. Als nächstes wurden drei Use Cases durch die Testperson ausgeführt. Dabei mussten beispielsweise Filter- und Suchfunktionen angewendet werden. Abschließend wurden Mockups von zwei Versionen der Startseite, der Suchfunktion, der Benachrichtigungsfunktion, den Favoriten, Projektfortschrittsübersicht und Bilder von Kartenmarkern präsentiert.

Die Testpersonen kommunizierten ihre Präferenzen zu den Mockups.

8.3 Ergebnis

In Vorbereitung auf die Nutzertests wurde eine Liste von Hypothesen in Bezug auf Menüführung, Funktionalität, Sprache und Design der KiezRadar-App erstellt. Während der Tests mit Nutzer:innen wurden die Hypothesen durch gezielte Fragestellungen adressiert, um evidenzbasiert Hypothesen zu bestätigen oder abzulehnen. Im Folgenden findet sich eine Übersicht der aufgestellten Hypothesen und deren Evaluation:

Tabelle 30: Ergebnis der Hypothesenevaluation aus den Nutzer:innentests.

Hypothese	Ergebnis	Erklärung
Die Nutzer:innen wünschen eine Aufteilung in mehrere Kategorien in der	abgelehnt	Viele Nutzer:innen verstehen nicht intuitiv, was der Reiter "Eigene" bezeichnet. Manche Nutzer:innen

Hypothese	Ergebnis	Erklärung
»Entdecken« Ansicht (aktuell, beliebt, eigene).		finden die "Entdecken" Ansicht durch die mehreren Reiter überladen. Die Bedeutung von "Aktuell" und "Beliebt" wird richtig verstanden und als sinnvoll empfunden.
Die Nutzer:innen möchten in der «Entdecken-Ansicht» gerne Bilder zu den Beteiligungen und die wichtigsten Infos gezeigt bekommen.	bestätigt	Ein Galeriebild in der Listenansicht kommt bei Nutzer:innen gut an, genauso wie eine kurze Informationsübersicht.
Die Nutzer:innen möchten gerne schnell erfassen und angezeigt bekommen, welche Beteiligungen bei anderen Bürger:innen besonders beliebt sind.	eher abgelehnt	Einige Nutzer:innen wünschen sich eine Information zur Beliebtheit von Einträgen. Manche Nutzer:innen fühlen sich durch eine Beliebtheitsanzeige beeinflusst und bewerten diese deshalb negativ.
Nutzer:innen wünschen sich eine separate Startseite.	kein Ergebnis	Die Anzahl von interviewten Nutzer:innen, die sich eine separate Startseite wünschen und Nutzer:innen, die dies nicht wünschen, hält sich im Gleichgewicht.
Nutzer:innen wünschen sich eine separate Suchfunktion als eigenen Menüreiter.	bestätigt	Eine separate Suchfunktion wird dringend gewünscht.
Nutzer:innen wünschen sich Benachrichtigungen durch die App.	bestätigt	Die meisten Nutzer:innen finden Benachrichtigungen sinnvoll und würden diese nutzen.
Nutzer:innen wünschen sich eine separate Information über alle Projektfortschritte, nicht nur über ihre Favoriten.	abgelehnt	Die meisten Nutzer:innen halten eine unterteilte Anzeige der Projektfortschritte für überladen und wünschen sich eine vereinfachte Lösung.

Unabhängig von den Hypothesen wurden folgende Beobachtungen gemacht:

- Favoriten werden als sehr sinnvoll empfunden
- Kategorie-Kacheln sind erwünscht
- Icons der Menü-Reiter werden gut erkannt
- Persönliche Options-Einstellungen in den Benachrichtigungen sind mehrheitlich erwünscht
- In der Karte werden kleine Vorschaufenster für die einzelnen Pins gewünscht bzw. als sehr hilfreich eingestuft

Folgende neue Fragen sind durch die Evaluation entstanden:

- Sind Icons als Kartenmarker kontextuell verständlich oder sollten sie weggelassen werden?
- Wie kann die Filtereingabe funktional und übersichtlich dargestellt werden?
- Wie viele Informationen sollte die Listenansicht der Einträge enthalten?
- Inwiefern sollen Informationen in der Listenansicht durch Kacheln oder Icons dargestellt werden?

9 Weiteres

9.1 Zusammenarbeit

KiezRadar war von vornherein darauf angelegt, ein partizipatives Projekt zu sein, das heißt, Bürger:innen, Verwaltungsmitarbeiter:innen, Quartiersmanagement oder Informationsplattformen einzubinden.

Die Zusammenarbeit gestaltete sich einfacher und deutlich erfolgreicher als erwartet. Die Frage, ob eine App unter Beteiligung der Stakeholder agil im Verwaltungskontext erstellt werden kann, darf mit »ja« beantwortet werden.

Dabei hat geholfen, dass mit dem CityLAB ein sehr agiler Partner zur Verfügung stand, der durch Kontakte, Mitarbeit und Räumlichkeiten die Teilnahme verschiedener Personen ermöglichte. Leider wurden wir durch COVID-19 gezwungen, den zweiten Teil des Projekts online zu gestalten.

Das Projekt mein.berlin.de wurde über die Senatskanzlei eingebunden und lieferte gut aufbereitete Projekt- und Plandaten.

Das Quartiersmanagement Auguste-Viktoria-Allee war so freundlich, seinen Terminkalender zur Verfügung zu stellen. Leider konnten insbesondere die geplanten Testtermine mit dem Quartiersmanagement aufgrund von Corona nicht stattfinden, so dass die App lediglich gezeigt werden konnte, eine systematische Rückmeldung war im Projektrahmen nicht mehr möglich.

Die Bezirksverwaltungen stellen der Allgemeinheit die Veranstaltungskalender der Bezirksverordnetenversammlungen zur Verfügung. Mit Hilfe des Herstellers CC e-gov konnten wir das eingesetzte Produkt ALLRIS über dessen XML-Schnittstelle ansprechen.

Das Projekt suchte auch das Gespräch mit Vertretern der Bezirksverordnetenversammlungen. Die Gespräche waren sehr positiv, bei einer Fortführung kann das große Interesse an KiezRadar für eine politische Verankerung genutzt werden.

9.2 Mediale Präsenz

Am 12. Juni 2019 wurde das CityLAB Berlin eröffnet und das Projekt KiezRadar seitens der Projektleiterin Susanna Kuper im Rahmen einer Pitch-Session auf der Eröffnungsfeier vorgestellt. Vorab und während des Pitches wurde dieser über die Social Media Kanäle von Fraunhofer FOKUS, dem DPS-Newsletter und der Fraunhofer Webseite beworben:

- <https://twitter.com/fraunhoferfokus/status/1138866946894700547/photo/1>
- <https://www.fokus.fraunhofer.de/de/dps/egovnewsletter/juni2019>
- https://www.fokus.fraunhofer.de/de/fokus/news/kiezradar_citylab_2019_06

Anlässlich der Eröffnung des CityLAB Berlin erschien ein Beitrag in der taz, bei dem auch auf das Projekt KiezRadar eingegangen werden sollte. Die Interviewfragen der Journalistin Katharina Schmidt hatten wir ausführlich beantwortet. Leider wurde das Projekt KiezRadar bei der endgültigen Fassung nicht erwähnt.

- <https://taz.de/Eroeffnung-CityLab/!5600159/>

Der erste Bürger:innen-Workshop im CityLAB Berlin am 28. November 2019 sowie der zweite Workshop mit Berliner Verwaltungsmitarbeiter:innen wurden über die Social-Media-Kanäle von Fraunhofer FOKUS und dem CityLAB Berlin, den DPS-Newsletter, die Webseiten Fraunhofer FOKUS und dem CityLAB Berlin sowie die SmartCity Berlin-Seite beworben.

Ein Bericht über den erfolgreich absolvierten Bürger:innen-Workshop wurde auf der Webseite des Fraunhofer FOKUS veröffentlicht:

- <https://www.fokus.fraunhofer.de/de/dps/kiezradar>

In dem politisch-technischen Print- und Online-Verwaltungsmagazin Kommune 21 wurde der Leitartikel der Januarausgabe zum Thema »Bürgerpartizipation« von der Projektleiterin Susanna Kuper verfasst. In dem Beitrag geht sie ausführlich auf die Erprobung neuer Formate der Bürgerbeteiligung ein und stellt das Projekt KiezRadar ausführlich als Beispiel vor.

- https://www.kommune21.de/meldung_33137.html

Die Nutzertests fanden im Rahmen der Science Week in Zusammenarbeit mit dem CityLAB Berlin statt. Die Teilnahme wurde über verschiedene FOKUS-eigene Kanäle gestreut und auch auf der Webseite der Science Week sowie der KiezRadar Projekthomepage veröffentlicht.

- https://www.fokus.fraunhofer.de/de/dps/events/KiezRadar_testen
- https://www.fokus.fraunhofer.de/de/fokus/event/science_week2020
- <https://falling-walls.com/event/testing-kiezradar/>
- <https://kiezradar.fokus.fraunhofer.de/science-week/>

Für den Digital Social Summit 2021 wurde ein Kurzvideo zur Vorstellung des Projektes produziert und auf YouTube veröffentlicht.

- <https://twitter.com/fraunhoferfokus/status/1376493695990702082>
- <https://www.youtube.com/watch?v=PutuLi6SSw8&t=1s>

9.3 Open Source

Der Code für die API, das Backend sowie die App wurde auf gitlab unter folgenden Lizenzen veröffentlicht:

- KiezRadar-App
 - <https://gitlab.com/kiezradar/app>
 - GNU General Public License
- KiezRadar-Backend
 - <https://gitlab.com/kiezradar/backend>
 - Lesser GNU General Public License
- KiezRadar-Webseite
 - <https://gitlab.com/kiezradar/website>
 - GNU General Public License

Zusätzlich zum reinen Code gibt es Dokumentationen, beim Backend für Entwickler:innen; bei der App für Anwender:innen und Entwickler:innen.

Die Lizenzen besagen, dass der Code modifiziert oder weitergegeben werden darf, eine Veröffentlichung jedoch unter den gleichen Lizenzbedingungen erfolgen muss. Das garantiert, dass eine Weiterentwicklung wieder allen Entwickler:innen zu Gute kommt.

Das »Lesser« des Backends erlaubt die Nutzung des Backends auch für Closed-Source-Projekte oder für Open-Source-Apps mit anderer Lizenz. So kann das Backend frei genutzt werden, lediglich eine Weiterentwicklung müsste wieder unter gleicher Lizenz erfolgen.

Jedes Repository enthält einen Issue-Tracker, in dem bekannte Probleme bzw. zu erledigende Arbeiten bereits eingetragen sind. Die Nutzung der Tracker ist für alle möglich, so dass Entwicklungswünsche oder Probleme gemeldet werden können.

Die Repositories sind für Forks und Merge Requests freigeschaltet, so dass eine kontinuierliche Weiterentwicklung mit Hilfe der gitlab-eigenen Workflows betrieben werden kann.

10 Lessons Learned

10.1 Nutzerzentrierter Entwicklungsansatz

Der nutzerzentrierte Ansatz war ebenso wie die direkte Einbindung der Nutzer:innen ein sehr erfolgreiches Vorgehen. Durch die Workshops und Tests mit den Nutzer:innen konnten wertvolle Erkenntnisse und Impulse gesammelt werden. Dabei haben sich das systematische Vorgehen anhand von Hypothesen und der Methodeneinsatz des Lauten Denkens bewährt.

Die Organisation der durchgeführten Formate waren nicht zuletzt wegen der Corona-Pandemie mit hohem Aufwand verbunden. Abhilfe hätte hierbei die Bildung einer festen Gruppe von Ansprechpartner:innen schaffen können, die über den Projektverlauf bei Fragen zu Designentscheidungen oder für das Einholen von Feedback zu Mockups etc. konsultiert werden kann. Denkbar wäre hier auch ein Treffen in regelmäßigen Abständen abzuhalten. Eine solche Gruppe könnten auch für die gemeinsame Entwicklung von Designentwürfen genutzt werden.

10.2 Einbindung der Verwaltung

Ebenso erfolgreich war die Idee, die Verwaltung als wichtigen Stakeholder direkt in die Entwicklung einzubeziehen. Der dafür durchgeführte Workshop war erfolgreich und lieferte viele Erkenntnisse aus Sicht der Verwaltung, sowohl für Inhalte als auch den Umgang mit einer solchen App.

Leider konnte das volle Potential bedingt durch den Zwang zu einem Onlineworkshop nicht genutzt werden. Das Format sollte in Zukunft wiederholt und stärker genutzt werden.

10.3 Tests und Feedback

Trotz der Einschränkungen durch die Corona-Situation waren die Tests durchführbar und lieferten wertvolles Feedback. Gerade beim Design und der Bedienung der App waren die Nutzer:innen-Tests sehr hilfreich.

Geplant war eine Testreihe im CityLAB vor Ort, es wäre gut gewesen, dass auch so durchführen zu können. Gerade Tests mit Beobachter:innen sind vor Ort noch wertvoller, wenn die Haptik des Handys genutzt werden kann.

Unabhängig davon kann empfohlen werden, feste Test- bzw. Feedback-Gruppen zu etablieren und deren Feedback zu nutzen. Auch hier fehlte uns leider coronabedingt die Zeit dafür.

10.4 Qualität der Datenquellen

Im Projekt wurde auf bestehende Datenquellen zurückgegriffen, ohne während des Projekts Rückmeldungen zur Datenqualität zu liefern.

Es fiel auf, dass die Qualität der Daten sowohl innerhalb eines Systems aber vor allem zwischen den Systemen sehr unterschiedlich ist. Insbesondere die für KiezRadar notwendigen

Geoinformationen waren selten vorhanden, da in den Systemen eine Ortsangabe über einen Text ausreicht.

Folgende Datenverbesserungen werden vorgeschlagen und den Datenlieferanten im Nachgang des Projekts angepasst an ihre Datenmodelle mitgeteilt:

- konsequente Übergabe von Geoinformationen
- Titel und Beschreibungen mitgeben
- Zuordnung von Projekten und deren Teilstücken für Nachverfolgung
- Laufzeiten von Projekten
- eindeutige Kennzeichnung von Ereignissen
- Datum- und Zeitangaben normieren
- wiederkehrende Ereignisse kennzeichnen

10.5 Open Source

Im Projekt konnte die Open-Source-Stellung aufgrund der langwierigen Klärung der Rechtslage erst spät beginnen. Es kann nur empfohlen werden, die konkreten Bedingungen und Lizenzen für Open-Source frühzeitig zu klären und womöglich vertraglich zu fixieren. Im vorliegenden Fall waren alle Seiten über das Vorgehen und die Lizenzen einig, es fehlte lediglich die vertragliche Fixierung.

Von früher Etablierung von Open-Source-Prozessen profitieren insbesondere Dokumentation und Qualität.

10.6 Hybride vs. native App-Entwicklung

Eine hybride App-Implementierung erspart doppelte Arbeit bei dem Ausrollen einer App für verschiedene Betriebssysteme. Die Nachteile zu einer nativen App betreffen dabei einige wenige Aspekte, die sehr spezifische Nutzung bestimmter Sensoren und betriebssystemeigene Funktionen angehen. Für den Funktionsumfang einer App wie KiezRadar ist das jedoch nicht relevant.

Eine native App kann für einen Prototypen dennoch einen gewissen Mehrwert bringen, da man die ganze Breite der Funktionen und Sensoren nutzen kann. Ein Prototyp zeigt zunächst nur eine mögliche Implementierung und eventuell Grenzen des Machbaren. Eine Implementierung als rein native App in so einem Kontext ist in jedem Fall eine vertretbare Entscheidung. Jedoch ist die Nachnutzung des Programmcodes oder bestimmter Teile dessen dann nur eingeschränkt möglich, wenn man den Ansatz hin zu einer einheitlichen Code-Basis wechseln möchte.

Inzwischen ist durch beispielsweise das React-Native-Framework auf Basis von JavaScript eine Erstellung von nativen Apps mit einer einheitlichen Codebasis möglich. Somit kann eine App für verschiedene Betriebssysteme programmiert werden, ohne dabei verschiedene Programmiersprachen beherrschen und verschiedene Projekte verwalten zu müssen. Das bedeutet eine erhebliche Arbeitersparnis. Zudem können native Codeteile eingebunden werden, um zum Beispiel die Nachteile rein hybrider Apps zu umgehen.

10.7 Die Idee kommt an – Bedarf für die KiezRadar-App vorhanden

Oft ist es so, dass eine Idee theoretisch gut aussieht, in der Praxis jedoch keinen richtigen Bedarf erfüllt.

Zu unserer großen Freude war das bei KiezRadar anders: nicht nur waren alle bei den Workshops gern und gut beteiligt, auch das Ergebnis kommt gut an.

Die Gespräche mit Stakeholdern, wie bspw. Ansprechpartner:innen aus dem Quartiersmanagement, Vertreter:innen der BVV, Verwaltungsmitarbeiter:innen und Bürger:innen ließen einen klaren Bedarf für das Konzept der KiezRadar-App erkennen und der Prototyp erfüllt bereits die Erwartungen, obwohl die Rückkanäle noch nicht gebaut sind.

Der Mehrwert einer ergänzenden Anwendung zu den bestehenden Plattformen begründet sich u.a. durch die Zusammenführung mehrerer Informationsquellen, der adressatengerechten Aufbereitung, der Favoriten oder der Benachrichtigungsfunktion. Zudem zeichnet sich der Ansatz als mobile Anwendung durch den Vorteil des Standortbezugs aus.

Dadurch, dass Informationen auf mehreren Quellen zusammengeführt werden, profitieren verschiedene Akteure von einer derartigen App-Lösung. Gegenüber den bestehenden Plattformen hat KiezRadar einen breiteren konzeptionellen Fokus, der neben den im Prototyp genutzten Datenquellen eine flexible Erweiterung um weitere relevanten Datenquellen möglich macht.

11 Weiterentwicklung/Ausblick

Der aufgezeigte Bedarf für die KiezRadar-App unterstreicht, dass eine Fortführung des Projekts grundsätzlich sinnvoll ist. Mit dem bestehenden Funktionsumfang des Prototypens, den im Backlog gesammelten Anforderungen und der API kann für eine Fortsetzung auf umfangreichen Vorarbeiten aufgebaut werden.

Die aus unserer Sicht sinnvollen Zukunftsszenarien werden für Backend, Daten und App kurz aufgeführt.

11.1 Zukunft Backend

11.1.1 Was fehlt?

Im Backend fehlt zunächst die automatische Abfrage der Datenquellen. Sie ist Voraussetzung dafür, dass aktuelle, stabile Daten für die App zur Verfügung stehen. Die Automatisierung ist vorbereitet, bedarf jedoch noch der Überarbeitung und ausführlicher Tests.

Tests sind u. a. auch für die Härtung des Servers nötig, die für einen Dauerbetrieb durchgeführt werden müssen. Langfristig muss das Backend in eine Serverlandschaft integriert werden und gegen Angriffe gerüstet werden.

Zu guter Letzt fehlt noch der Rückkanal, der auch im Backend für größere Änderungen sorgen wird.

11.1.2 Optionen für die Zukunft

Die Weiterentwicklung und der Betrieb des Backends sind beispielsweise durch das CityLAB, Fraunhofer FOKUS oder das ITDZ möglich. Evtl. kann das Backend auch in mein.berlin.de integriert werden. Mischformen sind ebenfalls möglich.

11.1.3 Was bringt das?

Ein stabiles Backend ist die Grundlage für den Weiterbetrieb von KiezRadar, denn nur so können stabile und qualitativ hochwertige Daten von KiezRadar zur Verfügung gestellt werden.

Da die Schnittstelle offengelegt ist und auch die Daten öffentlich sind, wird eine offene Schnittstelle für jede:n geschaffen mit der Möglichkeit, eigene Anwendungen für die Daten zu entwickeln.

So kann gezeigt werden, wie öffentliche Daten durch Neuorganisation und Anreicherung verbessert und in anderen Kontexten genutzt werden kann. Beteiligung wird einfacher und besser ermöglicht.

11.2 Zukunft Datenaufbereitung

11.2.1 Was fehlt?

Auf technischer Ebene fehlen einheitliche Mindestdaten der Datenquellen. Dies wird im Projekt adressiert und erste Verbesserungen werden zurückgemeldet. So wird die Datenqualität und damit auch die Nutzung öffentlicher Daten verbessert.

Die Daten sollten neben der Standardisierung auch noch um weitere Inhalte angereichert werden, z. B. um:

- Projektinformationen ereignisübergreifend zu vernetzen,
- Projektfortschritte den Beteiligten zuordnen zu können oder
- redaktionelle Inhalte, die händisch für die jeweiligen Maßnahmen erstellt werden.

Des Weiteren fehlen noch weitere Datenquellen, hier gilt: je mehr, desto besser. Neben den großen Datenlieferanten wie z. B. Bezirksverordnetenversammlungen sollte es kleineren Initiativen ermöglicht werden, schnell und unkompliziert Daten beizutragen. Durch KiezRadar wird dann der Nutzen direkt sichtbar.

Mögliche neue Datenquellen sind z. B.:

- alle Ratsinformationssysteme
- alle vorhandenen Veranstaltungskalender der Quartiersmanagements Berlins
- Baustellenanzeiger »Hier Baut Berlin« (<https://blog.hierbautberlin.de/>, im Aufbau befindlich)

11.2.2 Optionen für die Zukunft

Für die Datenaufbereitung ist die Weiterentwicklung der Datenadaptores und die gezielte Ansprache von Datenlieferanten notwendig.

Zusätzlich kann ein Prozess etabliert werden, wie kleine Initiativen ohne Online-Kalender Daten zu KiezRadar beitragen können.

11.2.3 Was bringt das?

Die Daten sind das Rückgrat in der Akzeptanz der Anwendung. Mehr Daten bedeuten nicht nur mehr Ereignisse, sondern mehr Möglichkeiten der Beteiligung, einfacheren Zugang zu Verwaltungsvorgängen und langfristig mehr Transparenz der Verwaltung.

Als »Nebeneffekt« steigt die Datenqualität der Anwendungen an, da mindestens die für KiezRadar notwendigen Datensätze mitgeliefert werden, von denen auch andere Anwendungen profitieren können.

Gleichzeitig gibt es mit KiezRadar einen Anwendungsfall, wie öffentliche Daten abgerufen, aufbereitet und genutzt werden können. Auch hier gilt: je mehr Datenquellen beteiligt sind, desto größer wird der Wissensschatz über deren Nutzung.

Zu guter Letzt wird auch der Verwaltungsseite die Scheu genommen, Daten bereitzustellen und »aus ihrer Kontrolle« zu geben, wenn deutlich wird, wie Daten positiv genutzt werden können.

11.3 Zukunft App

11.3.1 Was fehlt?

Derzeit fehlen etliche Funktionen, die bereits als Anforderungen beschrieben wurden. Das betrifft insbesondere alle Funktionen des Rückkanals, der noch nicht implementiert ist.

Beispielhaft sollen für die Erweiterung des Funktionsspektrums genannt werden:

- Etablierung eines Rückkanals (von den Bürger:innen zu der Verwaltung),

- standortbasierte Nachverfolgung von passierten Ereignissen,
- Durchführung kleinerer Umfragen über die KiezRadar-App.

Derzeit liegt die App als Android-App vor, es fehlen WebApps oder eine iPhone-Variante.

11.3.2 Optionen für die Zukunft

Auf Basis der Daten können unterschiedliche App-Formate für unterschiedliche Plattformen entwickelt werden. Es ist auch möglich, eine spezielle Nutzung der Daten zu implementieren, die im derzeitigen Konzept noch nicht bedacht wurde.

All das ist mit KiezRadar möglich, eine Entwicklung kann durch gezielte Projekte oder Förderung unterstützt werden. Das kann innerhalb des CityLAB stattfinden, durch Entwicklungsprojekte oder Wettbewerbe.

Die Öffentlichkeit der Entwicklung ermöglicht es dabei, auch außerhalb von Fraunhofer FOKUS Apps zu entwickeln, zu pflegen oder zu erweitern.

11.3.3 Was bringt das?

KiezRadar ist bereits eine gut konzipierte App für Bürgerbeteiligung, der allerdings noch einige Features fehlen. Das kann über gezielte und ungezielte Weiterentwicklung verbessert werden, so dass nicht nur eine gute App entsteht, sondern im Zuge dessen eine Referenz für die Nutzung des KiezRadar-Backends und von öffentlichen Daten im Allgemeinen.

12 Glossar, Abkürzungen

API	»application programming interface« – Anwendungsschnittstelle
fmpp	»FreeMarker-based file PreProcessor « – Präprozessor für Textdateien
ical	»iCalendar« – Format zur Speicherung von Kalenderdaten
json	»JavaScript Object Notation« – Format zur Darstellung hierarchisch strukturierter Daten
OParl	Standard der Open Knowledge Foundation für den Datenzugriff auf parlamentarische Informationssysteme
REST	»Representational State Transfer« – Paradigma für Webservices
SQL	Sprache zur Definition von Datenbankstrukturen
UX	»User Experience«
XML	»Extensible Markup Language« – Format zur Darstellung hierarchisch strukturierter Daten