

SynergyHub

Audit 4

1

- Willkommen zu unserer Präsentation zum Audit 4

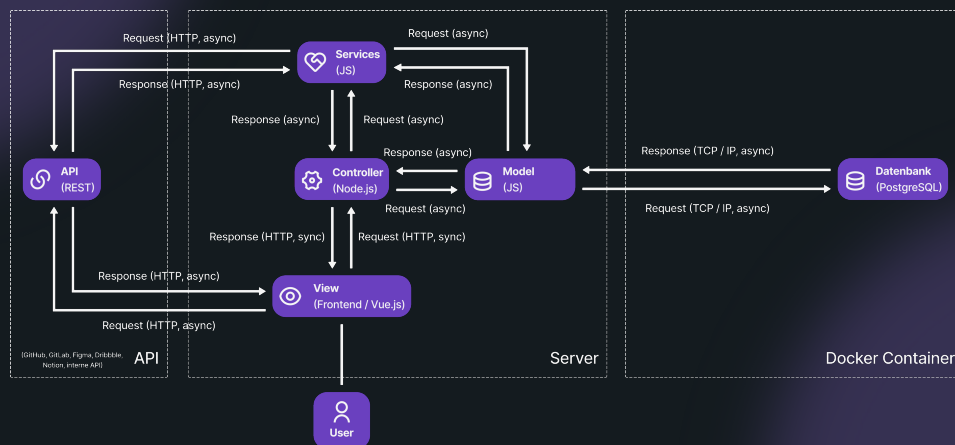
Inhalt

1. Architekturdiagramm
2. Main Prototype
3. Kritisch reflektiertes Prozessassessment
4. Ausblick

2

- **Folgende Inhalte:**
 - Architekturdiagramm
 - Main Prototype
 - Vorstellung
 - Code-Inspektion
 - Kritisch reflektiertes Prozessassessment
 - Ausblick

Architekturdiagramm



3

- **Architekturdiagramm:**

- Spiegelt die MVC-Architektur wider
- API-Aufrufe sowie Anfragen von der View zum Controller sind
- Um alle Clients über Aktualisierungen zu benachrichtigen wird SocketIO verwendet
- Da Projekt nicht deployed wurde, befindet sich Datenbank in einem Docker Container
- Außerdem wurde NGROK verwendet, da die Webhooks eine gültige Domain erwarten und diese sonst nicht ohne eine gültige URL laufen würden
 - NGROK wird außerdem verwendet, da sonst Probleme mit der Same-Origin-Policy entstehen

Main Prototype

Vorführung und Code-Inspektion

4

- Nun kommen wir zum Main Prototype
- Hier hat sich v.a. in der Architektur im Vergleich zum Rapid Prototype
- Genaueres kann anhand des Codes gezeigt werden

Kritisch reflektiertes Prozessassessment

5

- Nun soll kurz der Prozess kritisch reflektiert werden

Evaluation der Zielsetzungen

Zusammenführung von verschiedenen Arbeitsbereichen in „Collections“	→	Wurde weitgehend erfüllt, nur Dokumentation noch nicht wirklich integriert
Integration von standardisierten Plattformen	→	Wurde erfüllt, kann aber in Zukunft noch mit weiteren Plattformen ergänzt werden
Schnellere Projektabwicklung	→	Durch die Übersicht und Transparenz durch die Plattform werden Arbeitsprozesse beschleunigt
Social-Project	→	Wurde verworfen, da sich Fokus von Projekt gewandelt hat
Benachrichtigung aller Teilnehmer bei Änderungen	→	Wurde erfüllt und trägt zur Transparenz des Projektfortschrittes bei

6

- **Evaluation der Zielsetzung:**
 - Ziele wurden weitgehend erreicht, bzw. zum Teil auch im Laufe des Projektes an den sich ändernden Fokus des Projektes angepasst

Analyse des Prozesses

- Umfassender und gut strukturierter Projektdurchführungsprozess
- Entscheidende Schritte: Suche nach externen Diensten, Auswahl von Frameworks, Proof of Concepts
- Systematische und effiziente Entwicklung und Implementierung
- Auftreten von Bugs während der Implementierung, erforderte schnelle Behebung
- Hauptfokus auf Backend-Entwicklung, Vernachlässigung des Frontend-Designs
- Einschränkung: Fehlen eines responsiven Designs
- Gerade am Ende Konzentration auf Bugfixes und Lighthouse-Score

7

- **Analyse des Prozesses:**

- Gute Struktur bei Durchführung durch Planung und Sicherstellung der Umsetzbarkeit des Vorhabens
- Durch Auftreten von Bugs wurde Zeitplan etwas verengt
- Dadurch nicht ausreichend Zeit, das Frontend komplett umzusetzen
- Gegen Ende war es wichtig, einige Fehler, die auftraten zu beheben
- Um die Arbeit von Menschen mit Einschränkungen zu erleichtern, wurde sich auch stark mit den Google-Lighthouse-Anforderungen befasst

Kritische Reflexion

- Ursprüngliche Ziele wurden nur größtenteils Erreicht / Änderung der Ziele
- Einschränkungen durch externe Dienstleister, insbesondere bezüglich Antwortzeiten
- Geringfügige Beeinträchtigung der Benutzererfahrung, jedoch keine signifikanten Auswirkungen auf Gesamterlebnis der Plattform
- Notwendigkeit einer Überarbeitung des Frontends für bessere Nutzerfreundlichkeit und Bedienbarkeit, einschließlich responsivem Design
- Erforderlichkeit früherer Maßnahmen zur Konfliktlösung innerhalb der Gruppe hervorgehoben

8

- **Kritische Reflexion:**
 - Ursprünglichen Ziele wurden nicht vollständig erreicht, was zusammenhängt, dass der Fokus des Projektes verschoben wurde
 - Anderen Punkten sind auf der Folie zu sehen

Lessons Learned

- Wichtigkeit der frühzeitigen Erkennung und Bewältigung technischer Herausforderungen
- Besonderes Augenmerk auf Integration verschiedener Plattformen und nahtlose Zusammenarbeit gelegt
- Empfehlung für zukünftige Projekte: Gründlichere Vorabprüfung technischer Hürden
- Hohe Anforderungen an Zeit- und Ressourcenmanagement
- Notwendigkeit realistischer Zeitpläne und effizienter Ressourcennutzung
- Ziel: Gewährleistung des Projektfortschritts und Minimierung potenzieller Verzögerungen
- Früheres Befassen mit Konflikten im Projekt-Team

9

- **Lessons Learned:**

- Herausforderungen sollten so früh wie möglich erkannt und bewältigt werden
- Zudem sollte ein realistischer Zeitplan erstellt werden, der auch die Risiken beinhaltet
- Außerdem wollen wir uns in Zukunft früher mit Konflikten innerhalb der Gruppe befassen

Fazit

- Erfolgreiche Integration und Zusammenarbeit durch Collections
- Beschleunigung des Projektfortschritts
- Wichtige Lehren: frühzeitige Erkennung und Bewältigung technischer Herausforderungen, effizientes Zeit- und Ressourcenmanagement
- Projekt hat Ziele erreicht und wertvollen Beitrag zur Verbesserung der Projektarbeit geleistet
- Plattform bietet innovative Lösung für Projektzusammenarbeit mit Potenzial, die Art und Weise, wie Projekte durchgeführt werden, zu verändern

10

- **Fazit:**
 - Allgemein lässt sich sagen, dass das Projekt seine Ziele erfüllt
 - Außerdem kann man einige Lehren aus diesem Projekt für zukünftige Projekte ziehen

Ausblick

- Überarbeitung der UI / UX
- Responsive Design
- Mehr Anpassbarkeit, v.a. in den Benachrichtigungseinstellungen
- Möglichkeit Tasks zu bearbeiten
- Effizienterer Code durch Evaluation
- Wirkliche Aufteilung der Services
- Statt einen API-Controller zu verwenden, einen Service daraus machen

11

- **Ausblick:**

- Es gibt einige Aspekte, die noch nicht im Prototyp umgesetzt werden konnten
- Da dieses Projekt in Zukunft außerhalb des Moduls noch weiter ausgebaut werden soll, werden diese Aspekte noch Einzug in die Plattform finden