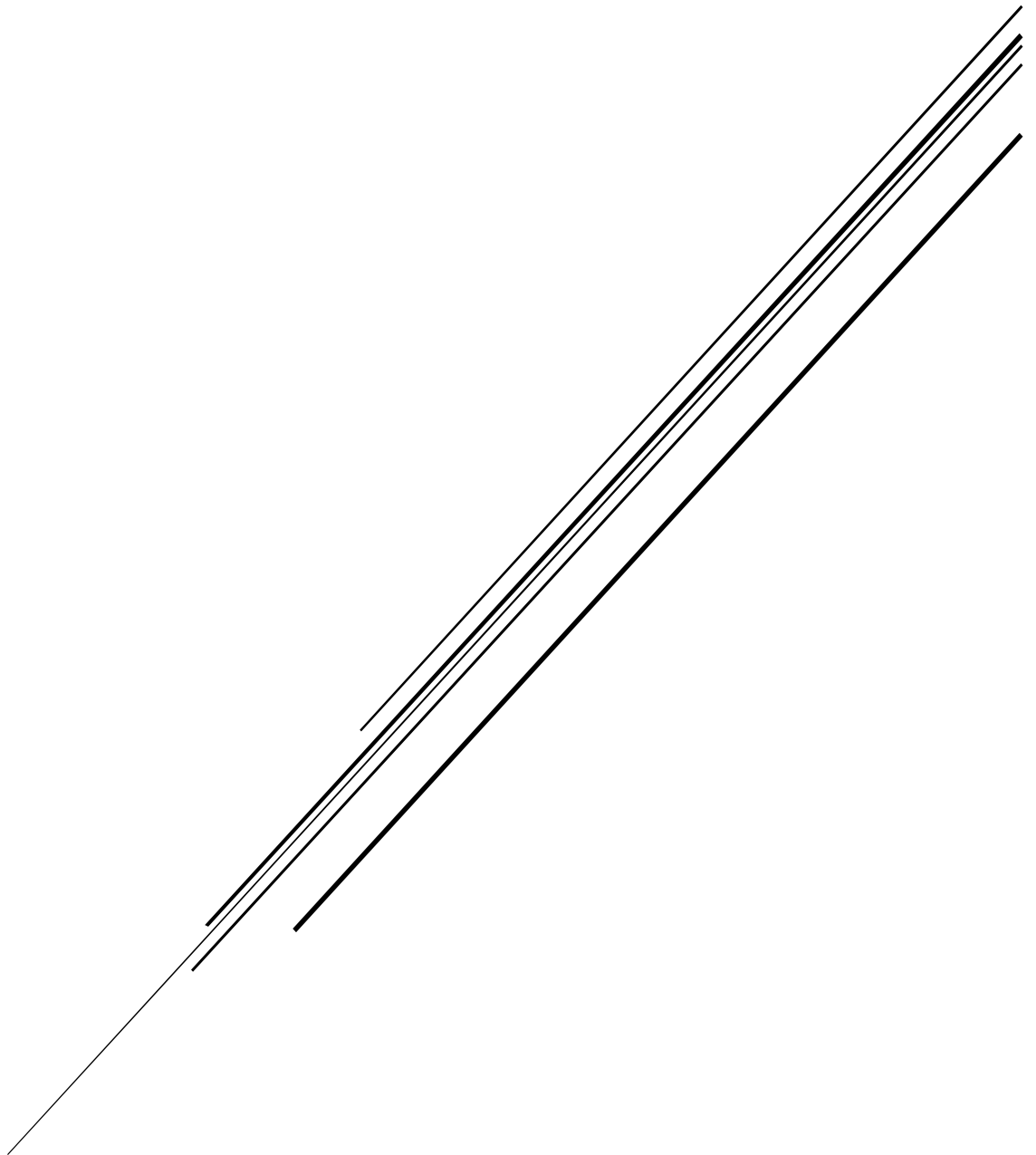


PROJEKTDOKUMENTATION

SAVEUP-APP

Ilia Kalygin



Modul 335
18.06.2024

Ausgangslage

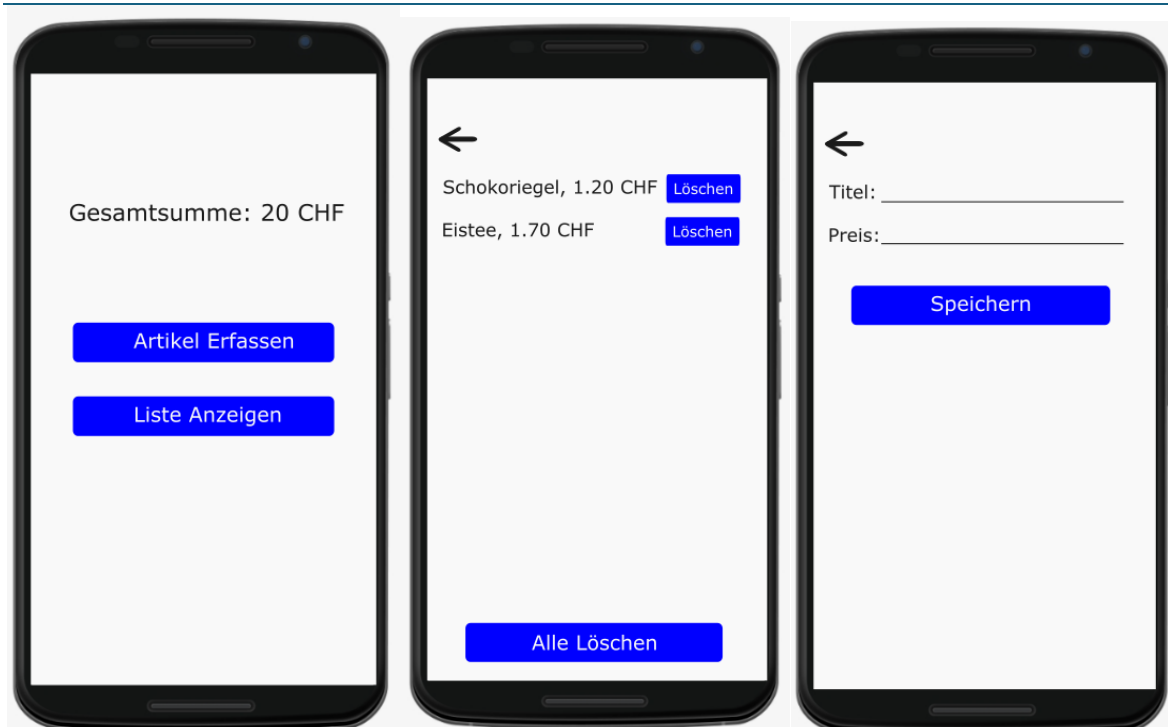
Für eine grössere private Investition, wie zum Beispiel eine Urlaubsreise, soll gespart werden. Um dies zu erreichen, werden übliche kleine Ausgaben wie Kaffee oder Süssigkeiten vermieden. Diese Einsparungen sollen in einer App festgehalten werden, damit der Benutzer jederzeit den angesparten Geldbetrag einsehen kann.

Ziele

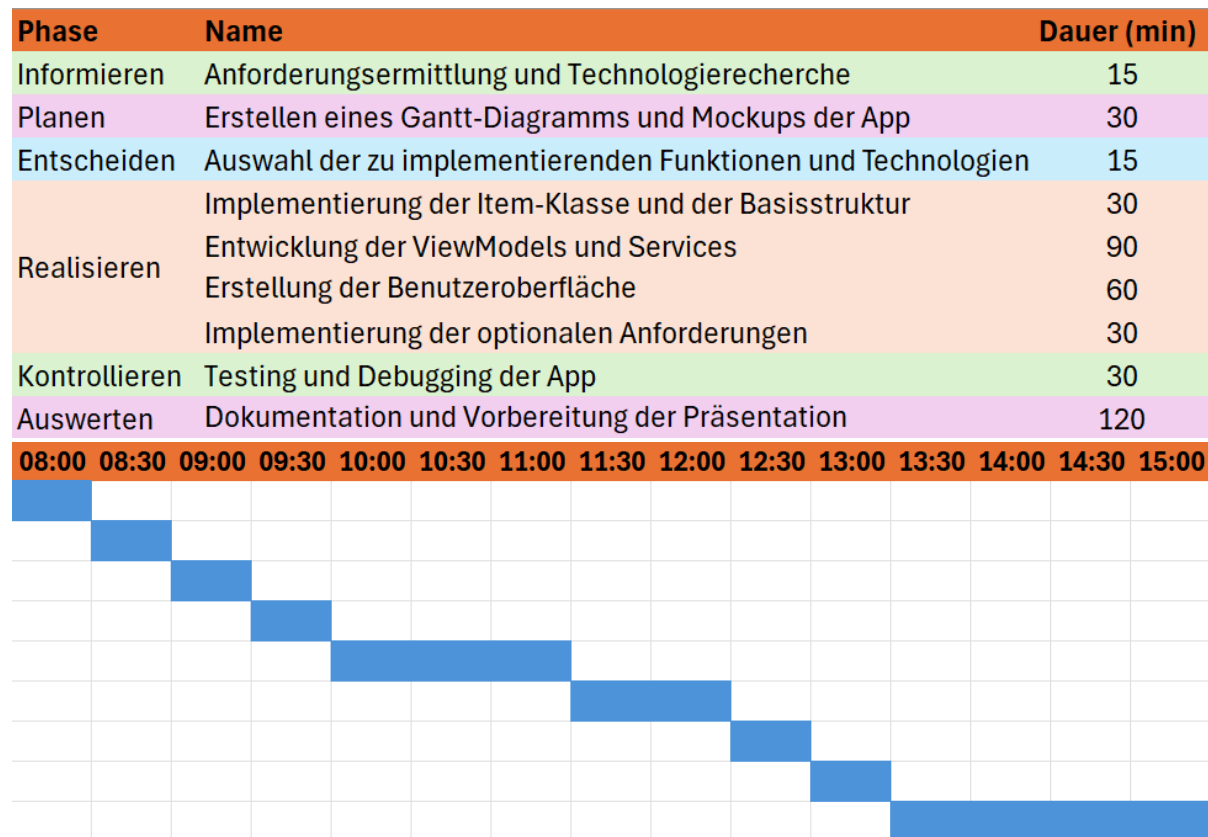
Das Ziel des Projekts ist die Entwicklung einer .NET MAUI App mit dem Namen "SaveUp", die folgende Funktionen bietet:

- Erfassen und Speichern von gesparten Verzichtsprodukten mit Kurzbeschreibung und Preis.
- Anzeige der erfassten Produkte in einer Liste mit der Summe der gesparten Beträge.
- Einfache und intuitive Bedienung mit ansprechendem Layout.
- Codestrukturierung nach dem Model-View-ViewModel (MVVM) Entwurfsmuster.
- Verwendung von XAML-Styles für die Steuerelemente.
- Implementierung eines eigenen App-Icons.
- Umfassende Dokumentation und Testing der App.

Mockups



Planung nach Gantt



Testplan und Protokoll

Testplan

1. Unit Tests:

- Testen der Item-Klasse auf korrekte Speicherung und Rückgabe von Attributen.
- Testen der ItemService-Klasse auf korrekte Speicherung, Laden und Löschen von Einträgen.

2. Integrationstests:

- Sicherstellen, dass die ViewModels korrekt mit den Services interagieren.
- Überprüfung der Datenbindung zwischen ViewModels und Views.

3. GUI Tests:

- Testen der Benutzeroberfläche auf verschiedene Eingaben und Aktionen.
- Überprüfung der Navigation zwischen den Seiten.

Testprotokoll

- **Testfall 1:** Erstellen eines neuen Eintrags
 - **Beschreibung:** Ein neuer Eintrag wird erstellt und gespeichert.
 - **Erwartetes Ergebnis:** Der Eintrag erscheint in der Liste und der gesparte Betrag wird aktualisiert.
 - **Ergebnis:** Erfolgreich
- **Testfall 2:** Löschen eines Eintrags
 - **Beschreibung:** Ein vorhandener Eintrag wird gelöscht.
 - **Erwartetes Ergebnis:** Der Eintrag wird aus der Liste entfernt und der gesparte Betrag wird angepasst.
 - **Ergebnis:** Erfolgreich
- **Testfall 3:** Navigation zwischen den Seiten
 - **Beschreibung:** Der Benutzer navigiert zwischen der Hauptseite, der Hinzufügen-Seite und der Listen-Seite.
 - **Erwartetes Ergebnis:** Die Navigation erfolgt reibungslos und die Daten werden korrekt angezeigt.
 - **Ergebnis:** Erfolgreich
- **Testfall 4:** Speichern der Einträge in einer JSON-Datei
 - **Beschreibung:** Die Einträge werden in einer lokalen JSON-Datei gespeichert und beim Neustart der App wieder geladen.
 - **Erwartetes Ergebnis:** Die Einträge sind nach dem Neustart der App weiterhin vorhanden.
 - **Ergebnis:** Erfolgreich
- **Testfall 5:** Optionale Anforderungen
 - **Beschreibung:** Implementierung und Testen der optionalen Anforderungen wie das Speichern in einer Backend-Datenbank und die grafische Darstellung der Daten.
 - **Erwartetes Ergebnis:** Die optionalen Funktionen arbeiten wie erwartet.
 - **Ergebnis:** Erfolgreich