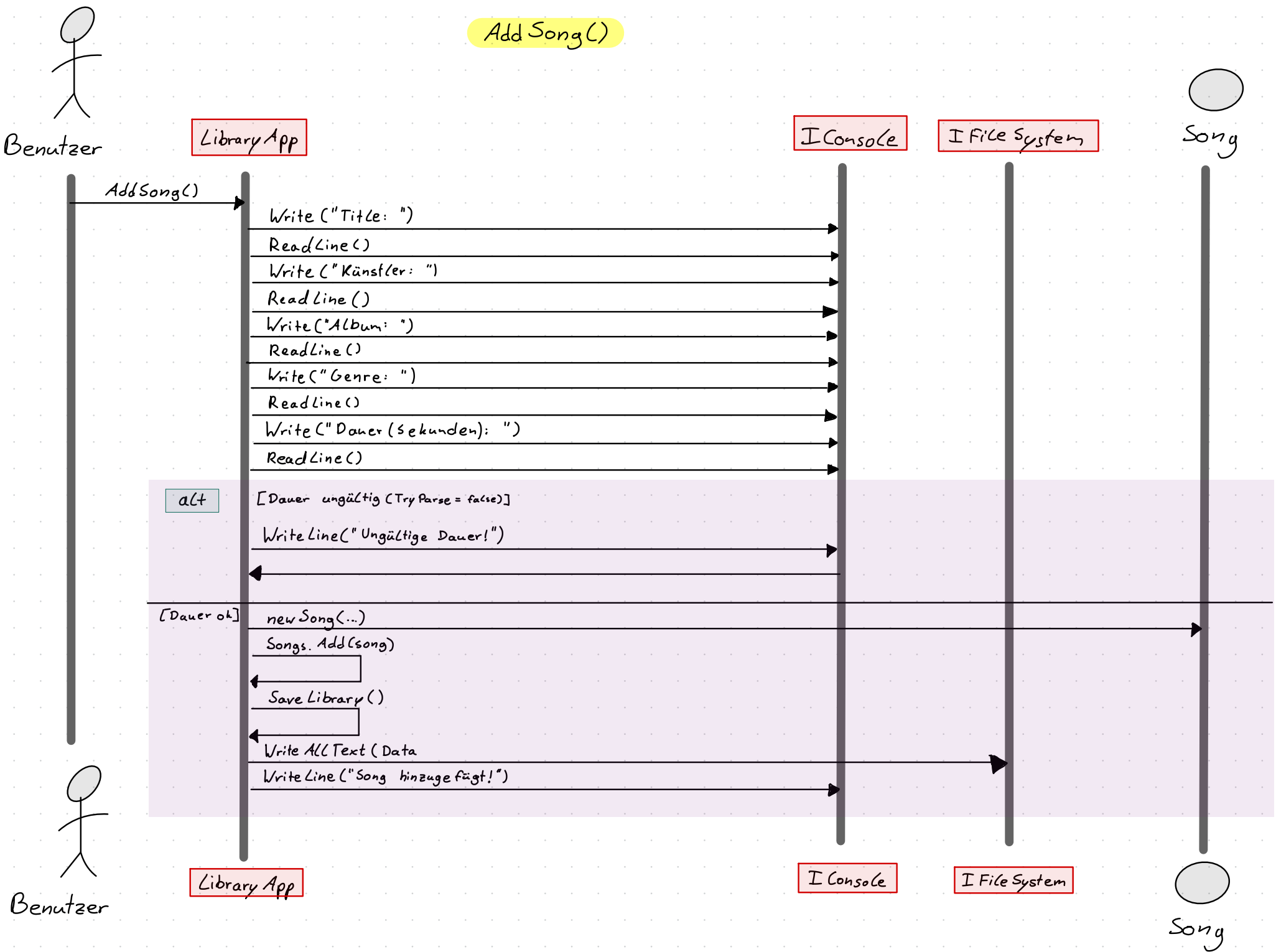
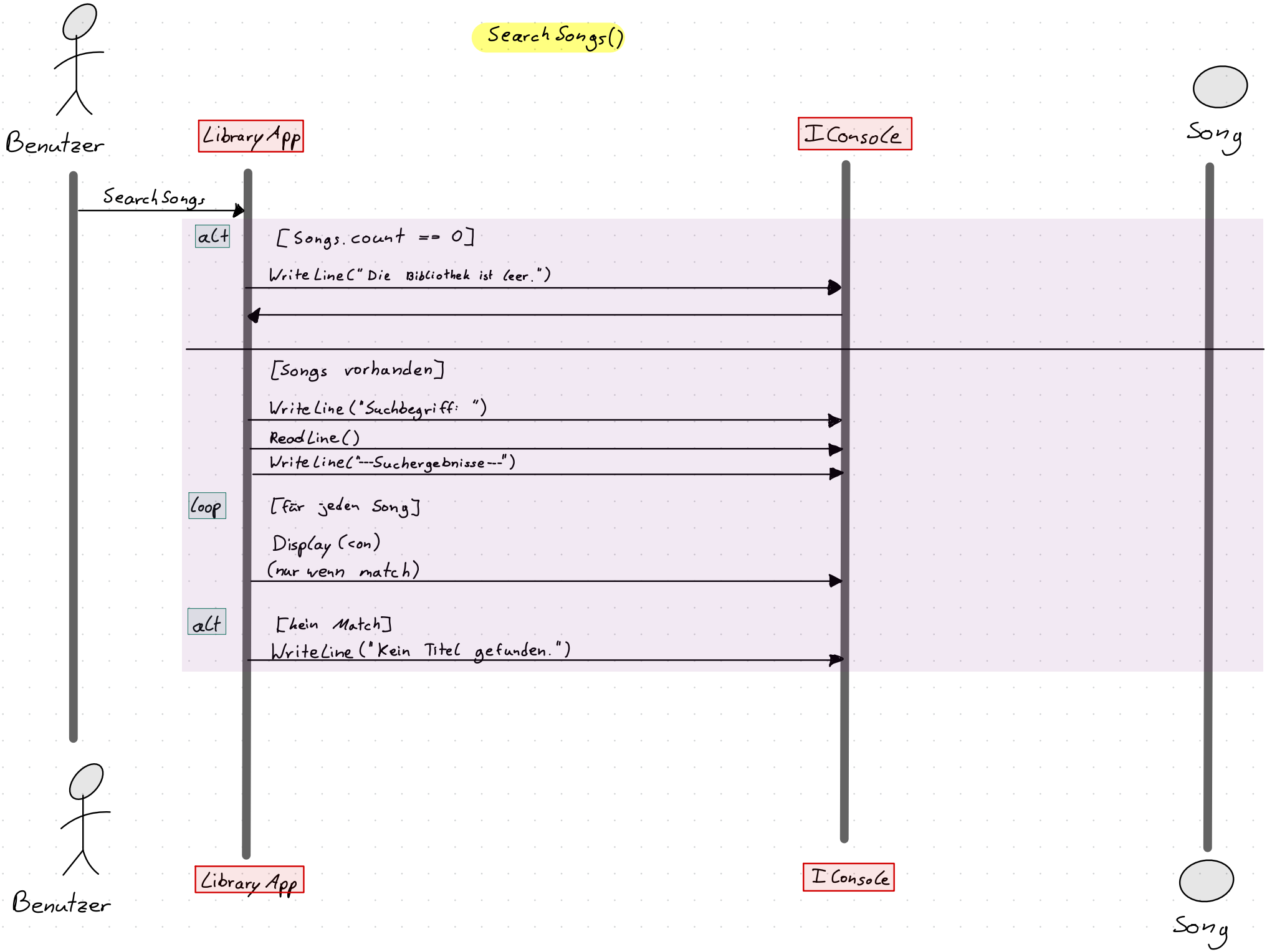
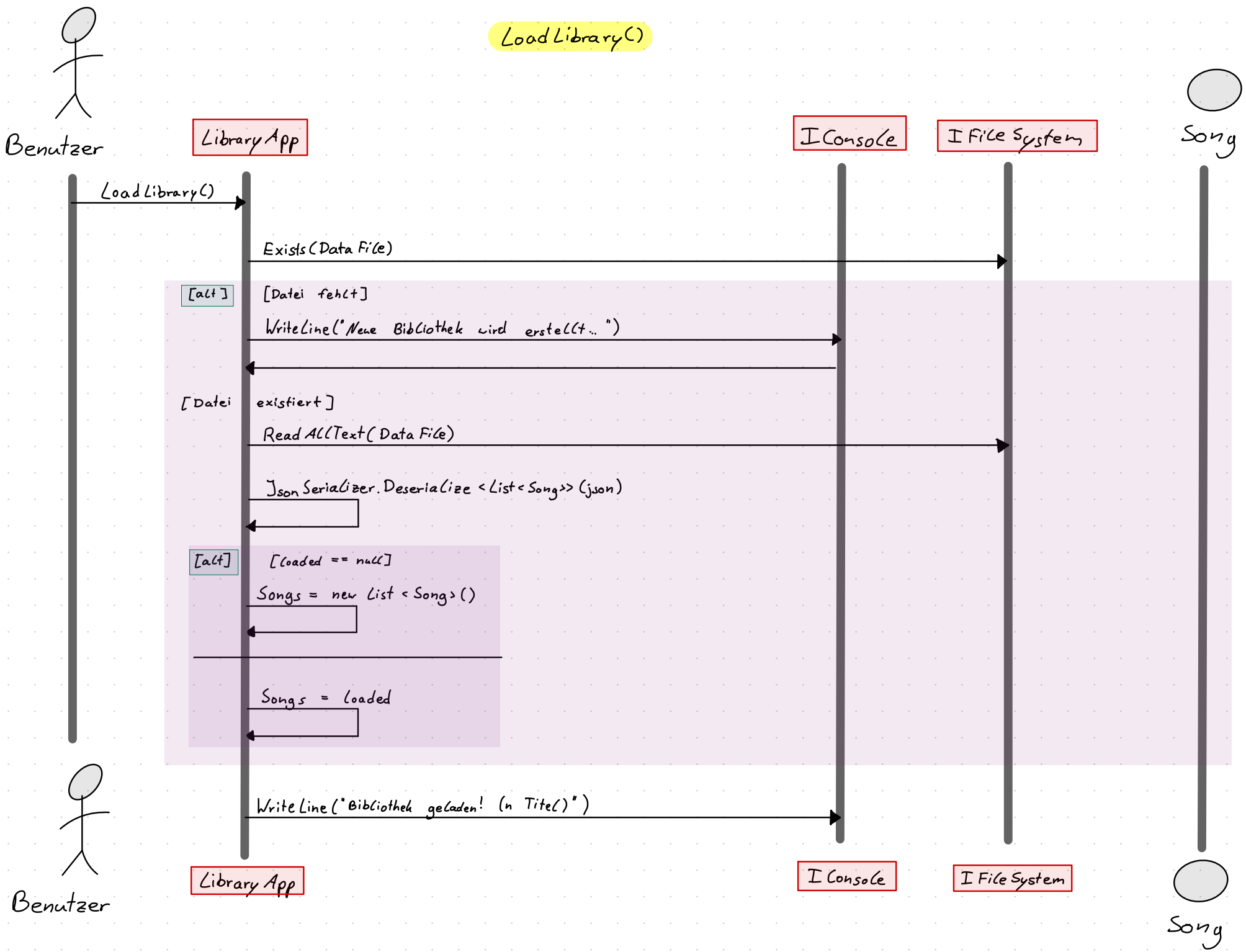
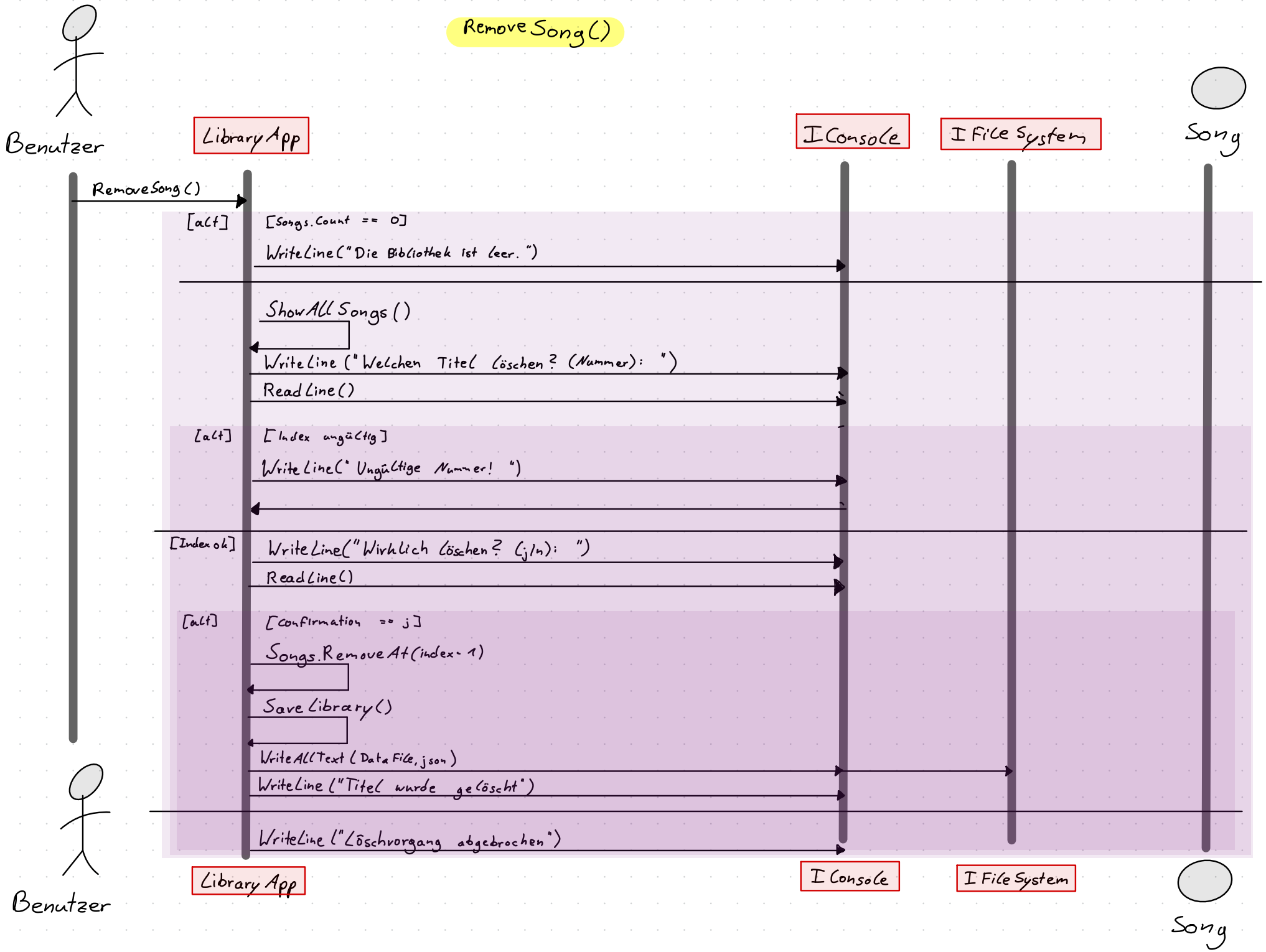

Add Song()
Benutzer
Library App
IConsole
IFile System
Song
AddSong()
Write ("Title: ")
ReadLine()
Write ("Künstler: ")
ReadLine()
Write ("Album: ")
ReadLine()
Write ("Genre: ")
ReadLine()
Write ("Dauer (Sekunden): ")
ReadLine()
alt
[Dauer ungültig (TryParse = false)]
WriteLine("Ungültige Dauer!")
[Dauer ok]
new Song(...)
Songs.Add(song)
SaveLibrary()
WriteAllText(Data
WriteLine("Song hinzugefügt!")
Benutzer
Library App
IConsole
IFile System
Song


SearchSongs()
Benutzer
LibraryApp
IConsole
Song
SearchSongs
alt
[Songs.count == 0]
WriteLine("Die Bibliothek ist leer.")
[Songs vorhanden]
WriteLine("Suchbegriff: ")
ReadLine()
WriteLine("---Suchergebnisse---")
loop
[für jeden Song]
Display(con)
(nur wenn match)
alt
[kein Match]
WriteLine("Kein Titel gefunden.")
Benutzer
LibraryApp
IConsole
Song


LoadLibrary()
Benutzer
LibraryApp
IConsole
IFileSystem
Song
LoadLibrary()
Exists(DataFile)
[alt]
[Datei fehlt]
WriteLine("Neue Bibliothek wird erstellt...")
[Datei existiert]
ReadAllText(DataFile)
JsonSerializer.Deserialize<List<Song>>(json)
[alt]
[loaded == null]
Songs = new List<Song>()
Songs = loaded
WriteLine("Bibliothek geladen! (n Tite()")
Benutzer
LibraryApp
IConsole
IFileSystem
Song


RemoveSong()
Benutzer
LibraryApp
IConsole
IFileSystem
Song
RemoveSong()
[alt]
[Songs.Count == 0]
WriteLine("Die Bibliothek ist leer.")
ShowAllSongs()
WriteLine("Welchen Titel löschen? (Nummer): ")
ReadLine()
[alt]
[Index ungültig]
WriteLine("Ungültige Nummer!")
[Index ok]
WriteLine("Wirklich löschen? (j/n): ")
ReadLine()
[alt]
[Confirmation == j]
Songs.RemoveAt(index-1)
SaveLibrary()
WriteAllText(DataFile, json)
WriteLine("Titel wurde gelöscht")
WriteLine("Löschvorgang abgebrochen")
Benutzer
LibraryApp
IConsole
IFileSystem
Song