

Masolism

Masolism --Aufbau

Es gibt eine klare Projekt und Datei Struktur. Jeder Befehl zu dem erstellten Projekt muss Folgend aufgebaut sein:

‚Projektname‘ , , ,—Thema‘ , -optionen‘

‚--Thema‘ sollte einen passenden Titel für das Projekt sein.

‚-optionen‘ sollten passende Eingaben haben um gut in der Kommandozeile im Projekt Navigieren kann.

- In diesem Projekt benutzen wir den Projektnamen Masolism, als Zusammensetzung aus den Wörten ‚Masochismus‘ und ‚Sozialismus‘, da es ein großes Projekt sein soll, welches Masochismus gebrauchen kann und Sozialismus verinnerlicht hat. Masolismus ist der Name der Community, masolism ist der Befehl um das Projekt anzusteuern.

Masolism --Inhalt

- Server Tablet
 - Abstrakt + Zusammenfassung
 - Hardware
 - Software
- Client Handy
 - Abstrakt + Zusammenfassung
 - Hardware
 - Software

Masolism -Tablet_Server -a0

Abstrakt: Wir entwickeln ein mobiles Endgeraet, welches in etwa die Leistung eines „Nvidia Jetson nano“ hat.

Es muss ein eigenes, auf „Android open source project“, basierendes Betriebssystem entwickelt werden.

Es müssen mehrere open source protokolle gefunden und entwickelt werden, welche für folgendes genutzt werden:

Masolism -Server_Tablet -a1

Ein leistungsstarker ,mobiler Server‘ (kurz ,lmS‘), welcher mindestens 3 Services anbietet...

“ ...1. i2P Router

- muss in der Lage sein als i2P Router genutzt zu werden, mit einer Offiziellen Website für das Projekt.
- Diese sollte etwas wie „GitLabs“ sein. Ein Forum wo der Code besprochen wird und neues Vorgestellt wird.

”

Masolism -Server_Tablet -A2

Ein leistungsstarker ‚mobiler Server‘ (kurz ‚lmS‘), welcher mindestens 3 Services anbietet...

“ ...2. Ein Fediverse Server

- muss in der Lage sein als Fediverse Server zu fungieren.

Es werden Eigene Social media Plattformen gehostet, mit dem Cluster an Servern, welche zur Verfügung stehen.

- Diese Funktion muss standard mäßig Deaktiviert sein und mit genauer Erklärung, was mit dem Gerät passiert bei Anwendung.

”

Masolism -Server_Tablet -A3

Ein leistungsstarker ‚mobiler Server‘ (kurz ‚lmS‘), welcher mindestens 3 Services anbietet...

“ ...3. Ein remote Server

- muss in der Lage sein als Android Desktop remote zu Agieren für das Client_Handy.
- Diese Funktion muss standard mäßig Aktiviert sein und mit genauer Erklärung, was mit dem Gerät passiert bei anwendung.

”

Masolism -Server_Tablet -A4

Ein leistungsstarker ,mobiler Server‘ (kurz ,lmS‘), welcher mindestens 3 Services anbietet...

“ ...Braucht zudem eine eigen Entwickelte Hardware.
Diese muss zu 100% open source Entwickelt werden, inklusive
benanntes Betriebssystem, Platinen, Gehäusen, Protokollen, etc.

”

Masolism –Server_Tablet -S0

Software: Platz für genauere Planung

Masolism –Server_Tablet -H0

Hardware: Platz für genauere Planung

Masolism -Client_Handy -a0

Abstrakt: Wir entwickeln ein mobiles Endgeraet, welches wie ein Slim-Client Funktioniert.

Es muss ein eigenes, auf „Android open source project“, basierendes Betriebssystem entwickelt werden.

Es müssen mehrere open source protokolle gefunden und entwickelt werden, welche für folgendes genutzt werden:

Masolism -Client_Handy -a1

Ein eigen Entwickeltes mobiles Endgerät, welches mit einer custom eSIM-Karte als Empfänger von Informationen, aus der i2P verbindung. Ohne eSIM-Karte muss es mindestens 30GiB Festplattenspeicher und 4GiB Arbeitsspeicher besitzen und ein minimalistisches, funktionales Betriebssystem laufen, auf dem es möglich ist apk Programme laufen zu lassen.

Falls möglich: 2 Computer einbauen, einen Leistungsstärkeren für das offline Handy und einen Leistungsschwächeren für das online Handy

”

Masolism –Client_Handy -S0

Software: Platz für genauere Planung

Masolism –Client_Handy -H0

Hardware: Platz für genauere Planung