

# Einführung

## Copyright

**Der Wandel des Geldes** - Die Zukunft des Zahlungsmittels und ökonomische Transformation  
von Adam Art Ananda

(C) Copyright 2023 Adam Art Ananda. Alle Rechte vorbehalten.

## Motivation

Ich habe etwa 14 Jahre lang als Softwareentwickler für Banken in Frankfurt und Zürich gearbeitet und miterlebt, wie Banken mit unserem Geld spekulieren. Dies wird auch als die große Derivate-Blase bezeichnet, die irgendwann platzen wird. Wir haben es bereits bei der Insolvenz von Lehman Brothers und vor nicht allzu langer Zeit auf Zypern gesehen, wo Menschen enteignet wurden und wiederum andere, teilweise keine Gelder mehr über Bankautomaten abheben konnten.

Aus gesundheitlichen Gründen musste ich 2014, nach meinem zweiten Burnout, aussteigen. Ich habe mir geschworen, nie wieder für Profit zu arbeiten. Zu der Zeit passte es gut, in diesem Bereich zu arbeiten, da ich eine fünfköpfige Familie zu versorgen hatte, die Katzen nicht mitgezählt. Damals keimte bereits die Idee in mir auf, dass Handlungsbedarf besteht, allerdings wusste ich noch nicht genau wie oder was.

Zu dieser Zeit absolvierte ich nebenbei ein Studium für Human Computer Interaction Design an der Universität Rapperswil. Da ich zu der Zeit nach Dänemark gezogen bin und keine Lust mehr hatte, jedes Wochenende in die Schweiz zum Studieren zu pendeln, habe ich das Studium nach dem zweiten Semester abgebrochen. Der Schwerpunkt im dritten und vierten Semester lag zudem auf Grafikdesign, was ich bereits zuvor bei der SGD abgeschlossen hatte.

Dennoch setzte ich mein Studium in Dänemark fort. Es handelte sich um ein Fernstudium zum Thema User Experience (UX). Dort war einer der Dozenten Don Norman, Autor des Buches "The Design of Everyday Things". Von ihm bekam ich die Aufgabe, eine App zu entwerfen, mit der man ein zeitbasiertes Geld verwenden konnte, um Dinge zu bezahlen. Genau dieses Design verwende ich in Shift.

Zudem habe ich das Buch "2020 - Die neue Erde" von Jesus Bruder Bauchi gelesen. Ich habe Bauchi vor ein paar Jahren in Wien auf dem UBUNTU-Festival kennengelernt. Ich möchte nicht zu viel verraten, da ich euch empfehle, dieses Buch selbst zu lesen. Hier jedoch eine kurze Anekdote daraus.

Der Protagonist wacht auf Mallorca auf und wundert sich, wo all seine Freunde geblieben sind. Ihm fällt auf, dass alles grün ist, was er von Mallorca nicht kannte. Nach einem Spaziergang trifft er auf eine Frau, mit der er sich ausgiebig unterhält. Er erfährt, dass er anscheinend eine Zeitreise ins Jahr 2025 gemacht hat und sich vieles zum Besseren gewendet hat. Überall gibt es Obst und Gemüse, sodass der Hunger jederzeit gestillt werden kann. Das Flusswasser ist trinkbar. Es fahren Elektroautos, die während der Fahrt von Teslatürmen aufgeladen werden, sodass kein Tanken mehr nötig ist.

Schließlich fragt er die nette Frau, wie das alles möglich wurde und was mit den Regierungen und den Reichen passiert ist. Sie antwortet: "Wir haben ihnen unsere Macht nicht mehr gegeben und ihr Geld nicht mehr benutzt."

Diese Synchronizität führte mich in eine bestimmte Richtung, und ich fragte mich:

Warum habe ich es geschafft, als ehemaliger Schlosser bereits über 35 Jahre als

Softwareentwickler zu arbeiten?

Warum habe ich Grafikdesign studiert?

Warum habe ich zusätzlich Human Computer Interaction Design studiert?

Warum bin ich spirituell geworden, mache Yoga und gebe Tantra-Massage-Workshops?

Warum manifestiert sich vor meinen Augen eine Dystopie, die dem Buch 1984 ähnelt?

Warum hatte ich in Portugal Stress und bin zurück nach Berlin gekommen, obwohl hier die Regierung mit verfassungswidrigen Methoden gegen ihre Bürger vorgegangen ist, um ihnen eine Impfung zu verabreichen?

In diesem Buch wirst du erfahren, wohin mich all das geführt hat.

Zusätzlich gebe ich dir mit diesem Buch eine Anleitung, wie du die App weiterführen kannst, falls ich aus irgendeinem Grund nicht mehr in der Lage sein sollte. Die App ist Open Source und verfügt über einen Plugin-Mechanismus, sodass sie von jedem weiterentwickelt werden kann.

# Für wen ist dieses Buch geeignet?

Wenn du daran glaubst, dass wir Menschen die Schaffung von Geld selbst in die Hand nehmen sollten, anstatt den Zentralbanken zu erlauben, immer mehr Geld zu drucken und uns in eine Inflation zu treiben.

Wenn du nicht möchtest, dass alle deine Zahlungen von Behörden einsehbar sind. Hier geht es um Privatsphäre.

Wenn du verhindern möchtest, dass in deinem Land ein System wie das Social-Credit-System in China eingeführt wird. Damit ist gemeint, dass dir beispielsweise Geld abgezogen wird, wenn du in bestimmten Kreisen gesehen wirst, die von jemand anderem als "problematisch" eingestuft werden. Oder dass dein Konto gesperrt wird, weil du gegenüber einer einflussreichen Person unhöflich warst. Mit anderen Worten, um Diskriminierung zu verhindern.

Dieses Buch richtet sich auch an Softwareentwickler, die Plugins für die Shift-App schreiben möchten. Ich zeige dir Schritt für Schritt anhand eines Beispiels, wie das geht.

# **Für wen ist dieses Buch nicht geeignet?**

Für Personen, die fest daran glauben, dass der Euro seinen Wert behält und den Regierungen vertrauen, dass sie die Preise stabil halten, die stetige Staatsverschuldung abschaffen und die Steuern senken werden und uns auch nicht überwachen wollen.

# **Warum ich glaube, das die App uns helfen wird**

Am Ende des Buches habe ich eine Kurzgeschichte verlinkt. Ein kurzes Video über die Ansichten des Autors, was den Sinn des Lebens angeht.

Vor ein paar Jahren hat ein Bruder aus Chile die Geschichte am Lagerfeuer vorgelesen.

Als er fertig damit war habe ich angefangen zu lachen... sehr lange... und als ich mich erinnert habe was Mooji Baba, ein spiritueller Lehrer, ich glaube aus Portugal, mal geantwortet hatte, auf die Frage: "Mooji, was hast Du gemacht, als Du erkannt hast, das Du erleuchtet bist?" Und er antwortete, "ich habe laut gelacht", mußte ich wieder lachen und lachen und lachen.

Ja, für mich macht die Geschichte Sinn und wenn ich daran denke, das wir alle EINS sind, dann

macht es auch Sinn einfach mal anzufangen, etwas im Außen zu ändern, damit wir unsere Einstellung ändern um dann in der Lage sind, es im Innen auch zu tun.

Wie im Innen so im Außen, heißt es im Kybalion. Ich bin der Meinung, das wir diese Wahrheit auch umdrehen können.

Wie im Außen so im Innen.  
- Adam Art Ananda

Wir dürfen halt auch im Außen die Veränderungen vornehmen.

Ich hoffe, daß Dir dieses Buch gefallen wird, und wenn nicht, ist es auch nicht soo schlimm.  
- Sonnenstern (\* 1892 - † 1982)

## **Erlös dieses Buches**

Der Erlös dieses Buches fließt in den Verein CrowdWare, welcher damit Land erwerben wird, auf dem kleine Jurten, Teepees und Tiny-Häuser gestellt werden und den Menschen kostenlos zur Verfügung stehen, die das Land dort behüten, damit auch sie zu einem Queraussteiger werden können.

# Danksagungen

All die Menschen, die mich begleitet haben, haben mich zu dem gemacht, der ich heute bin. Selbst diejenigen, die ich eventuell böse in Erinnerung habe, halfen mir auf eine bestimmte Weise.

Ich meine die sogenannten Arschengel, die und frech nen Spiegel vorhalten.

Nicht das ihr denkt das ich mich nun aus dem Staub machen möchte, ich möchte euch einfach nur beispielhaft meine Dankbarkeit ausdrücken. Bitte nachahmen, ist keine CopyRight drauf.

Ich danke dem Peter, ganz herzlich, da er mich bereits seit einem Jahr kostenlos bei sich in einer Zweiraum-Wohnung wohnen läßt. Du weißt, warum ich hier bin.

Ich danke dem Edward, ganz herzlich, weil er mir gestern Geld angeboten hat, das ich ablehnen konnte.

Ich danke der Angela, weil sie mich motiviert hat, dieses Buch zu schreiben.

Ich danke der Toni, dass sie mich gestern Abend durch einen Prozess geführt hat.

Ich danke der Anja, das sie mich durch ihre Trennung von mir, Jahre später in die Selbstliebe gebracht hat.

Ich danke der Arantxa, das sie mich in dem Prozess meiner Trennung via Chat begleitet hat, um mich auf die Selbstliebe vorzubereiten.

Ich danke Andy, da er mich bewundert hat, wie lange ich mich auf dieses Projekt konzentrieren konnte, obwohl er ständig rein kam, und mich abgelenkt hat.

Ich danke meinen Tantra-Lehrern in Berlin, das sie mich rauswarfen, weil ich etwas sagen wir mal, ungeeignetes auf Facebook gepostet habe, was mir zeigt, das sie auch nur mit Wasser kochen.

Ich danke Andro dafür, das er anscheinend extra für mich das Chakrensystem, bzw. die Körpertypen nach Wilhelm Reich um den kompensierten Regiden erweitert hat (Insider Gag).

Ich danke Wilhelm dafür, das er nicht Sigmund Freud gefolgt ist, sondern seinen eigenen Weg gefunden hat.

Ich danke Bernd für seine Aufarbeitung von Wilhelm seinen Werken und seine persönliche Unterstützung.

Ich danke Silvio, für die Erfindung der Demmurage (Schwundgeld).

Ich danke dem Bürgermeister von Wörgl, weil er das Schwundgeld ausprobiert hat.

Ich danke Heinz für seine ständige Inspiration.

Ich danke Bob für den Song "One Day".

Ich danke Ralph für seine phylosophischen Betrachtungen, seine Dusche und seinen Kaffee und die Abschaffung der Harz4-Sanktionen und das er das BGE einführen wird, sobald wir in Deuschland wieder eine Verfassung haben.

Ich danke Bernd für seine Sturheit und dem Gradido.

Ich danke all den starken Frauen, denen ich das Herz brechen durfte um herauszufinden, das es auch anders geht.

Ich danke all den Kriegshetzern, die mir gezeigt haben, das man Kriege mit Geld finanzieren kann.

Ich danke Neil, für seine Gespräche mit Gott.

Ich danke Jesus für sein Buch "2020 - Die neue Erde", das mich in die Lage versetzt hat, Regnosen zu schreiben, mit denen ich unsere Zukunft manifestieren kann.

Ich danke der Bibliothekarin in Portugal, das sie mir "1984" als wertvolles Buch empfohlen hat.

Ich danke dem Autor von "1984" für seine Vorlage, die ich hiermit einfach mal umdrehe und aus einer Dystropie eine Utopie mache.

Ich danke Robert und Cris, für das Interview, das auf engeSCHENKt.tv laufen wird.

Ich danke Thomas, dafür, das er das Interview veröffentlichen wird.

Ich danke Michael, das er mich zum Klardenker hat werden lassen.

Ich danke meinem himmlischen Vater, das er mich zum Generalbevollmächtigten Finanzen gemacht hat.

Ich danke all den lieben Menschen, die mir geholfen haben dieses Buch auf 120 Seiten zu bringen, indem ich sie hier erwähnen kann, damit es ein Bestseller wird.

Ich danke Joe, der mich zum Freeman gemacht hat.

Ich danke Andy, für das Ei und für den Marsianen, denn Kartoffeln sind meine Lieblingspflanzen.

Ich danke Steve für das iPhone und seine Schlußrede vor der Uni.

Ich danke Budda für seine Beharrlichkeit.

Ich danke Domingos für seine Unterstützung in Monchique.

Ich danke den Autoren, die dieses Buch in ihre Muttersprache übersetzen um es allen Menschen zugänglich zu machen.

Ich danke der Druckerei, die diese Buch drucken wird.

Ich danke den Pflanzen, die mein Bewusstsein erweitert haben.

Ich danke den Politikern für das Kasperle-Theater.

Ich danke den ganzen Trollen auf FB für ihre werlosen Kommentare.

Ich danke MS-Media für all die Lügen.

Ich danke YouTube für das Zensieren der Fakten.

Ich danke Bill das er uns zeigt, was man mit Geld alles für einen Mist anstellen kann.

Ich danke John, Paul, George und Ringo für ihre tolle Musik.

Ich danke John für den Song "Imagine".

Ich danke allen Begleitern vom Workshop "Das befreite Herz", wo ich mal so richtig frei war und mich angenommen fühlte.

Ich danke Jaquelin für die tantrische Iniziiierung.

Ich danke meinem Ausbildungsleiter, der mir in jungen Jahren gesagt hat, ich wäre überheblich, was mich motivierte, über mich hinaus zu wachsen.

Ich danke Aneesha für die Transformation der Prana-Flow-Massage zu einem Werkzeug der Heilung.

Ich danke Regina, das sie mein Modell für die Prana-Flow-Massage war.

Ich danke all den Menschen, die zu mir gekommen sind und mir vertraut haben.

Ich danke Phillip für seine Herzengüte.

Ich danke Gita, dafür, das sie mich mit Phillip verbunden hat.

Ich danke Dave, das er den Verein zusammen mit mir gegründet hat.

Ich danke all den Menschen, die sofort erkennen, das wir einen Weg in die Freiheit gefunden haben und diese Erkenntnis mit anderen teilen.

Ich danke Michael, das er mich UBUNTU (kein Geld kein Tausch und kein Handel) etwas näher gebracht hat.

Ich danke Cris für sein Zitat, "Lasst euch nicht zu Feinden machen!".

Ich danke Martin für dieses tolle Zitat, das er mir vor einigen Tagen geliefert hat.

Wenn ich nun noch der Rainbow-Family danken würde, dann wird das Buch zu dick, also mache ich es in ein paar Sätzen.

Ich liebe euch. See you in 5

Sollte ich wen vergessen haben, dann bitte Bescheid geben.

Thank You, Danke, Merci, muchas gracias, dankon, mi estas dankema, Estoy agradecido.



# Inhaltsverzeichnis

<b>Vorwort</b>	18
Anmerkung der Redaktion	19
<b>Erklärungen</b>	20
<b>Warum brauchen wir noch eine Kryptowährung?</b>	21
Eigentlich brauchen wir gar kein Geld.	21
Spekulationen sollten verhindert werden	23
Das Horten sollte verhindert werden	24
Bezahlen ohne Internet	26
<b>2030 - Ein kurze Geschichte</b>	30
Am Montag in der Schule	33
Abends Zuhause	42
Abends bei Peter	46
Erik	49
<b>Architektur der App</b>	51
Front End	44
Aufbau	53
Storj und Public Keys	58
API	59

Business Logik	60
WebService	62
<b>Plugins</b>	68
Independent Musik	68
Paletten-Standorte	70
Standorte leerer Häuser	72
Chat	75
Sozial Media	76
Shop	78
Desktop	79
<b>Plugin Beispiel</b>	81
Basic	74
ShiftAPI	85
FriendApi	85
Theme	87
AutoSizeText	89
CircularValuePicker	83
StorjAPI	91
Feature Request	96
<b>Über den Autor</b>	98
<b>Bücher</b>	103
<b>Quellcode</b>	104

# Vorwort

Zunächst einmal handelt es sich bei diesem Projekt um ein NO-BUDGET-Projekt. Bitte entschuldige meine Rechtschreibung. Ich habe keinen Editor gefunden, daher können einige Fehler auftreten und meine Grammatik ist möglicherweise nicht die beste.

Solltest Du einen Fehler finden, dann kannst ihn gerne behalten.

Wenn du keine Bücher mit Fehlern lesen möchtest, ist dies möglicherweise nichts für dich. Gib es lieber jemand anderem, anstatt dich zu beschweren. Beschwerden helfen dir nicht und sie helfen mir nicht.

Aber wenn du dieses Buch als wertvoll erachtest, lade ich dich ein, eine Rezension bei Tolino zu schreiben und ein paar Sterne für die Bewertung zu hinterlassen, die anderen Menschen hilft, mein Buch zu finden, und mir hilft, mehr Bücher zu schreiben, und um überleben zu können.

# Anmerkung der Redaktion

Ich biete dir dieses Buch im Geiste des Geschenks an. Dieses Buch ist unter der Creative Commons-Lizenz lizenziert, die es dir ermöglicht, es für alle nichtkommerziellen Zwecke frei zu verwenden. Das bedeutet, dass du Auszüge aus dem Buch kopieren und in Blogs usw. verwenden kannst, solange du nichts verkaufst oder es als Werbematerial nutzt. Ich bitte dich jedoch darum, die Quelle zu zitieren, damit auch anderen Menschen mein Werk zugänglich ist. Weitere rechtliche Details findest du auf der Website von Creative Commons: Creative Commons <https://creativecommons.org>

Das Wesen von Geschenken liegt darin, dass das Gegengeschenk nicht im Voraus festgelegt ist. Wenn du dieses Buch erhalten hast oder es kostenlos verteilst, würde ich mich über ein freiwilliges Geschenk freuen, das deine Dankbarkeit und Wertschätzung zum Ausdruck bringt. Du kannst dies auch über die folgende Website tun: Meine Bücher <http://artoftouch.crowdware.at/book.html>

Ein großer Teil meines Wissens in diesem Buch wurde mir damals gegeben und ich gebe es hiermit an dich weiter.

# Erklärungen

Im Text findest Du einige Begriffe wie zum Beispiel 'Go', 'Kotlin', 'API', mit denen Du bestimmt nichts anfangen kannst.

Erklärungen hierzu findest Du am Ende des Buches im Glossar.

Sollte ein Begriff nicht erklärt worden sein, so empfehle ich dir ChatGPT eine künstliche Intelligenz oder eine Suchmaschine um Dir den Begriff erklären zu lassen.

ChatGPT: <https://chat.openai.com/>

Suchmaschine: <https://www.startpage.com/>

# Warum brauchen wir noch eine Kryptowährung?

## Eigentlich brauchen wir gar kein Geld.

Einmal folgte ich einer Einladung zu einem Vortrag über die Neuerfindung des Geldes. Unter den Rednern befand sich auch Johannes Stüttgen, ein Meisterschüler von Joseph Beuys, ehemaliger Kunstprofessor aus Deutschland.

Johannes zeigte auf einer Tafel, wie unsere Wirtschaft funktioniert: Es gibt einen Bedarf, es gibt Rohstoffe, und anderswo wird aus diesen Rohstoffen ein Produkt hergestellt, um den Bedarf zu decken.

Er fragte die Runde, was noch fehlte...

Bevor er sagen konnte, dass das Geld fehlte, unterbrach ich ihn und sagte: "Lassen Sie das mal einen Moment so stehen. Mehr brauchen wir gar nicht. Die Wirtschaft könnte auch ohne Geld funktionieren. Die Rohstoffe sind kostenlos, und das Produkt könnte man auch kostenlos

herstellen, zumindest nach der Philosophie der UBUNTU-Bewegung aus Südafrika."

Nach dem Vortrag setzten wir uns bei einem Bier zusammen und unterhielten uns.

Ja, derzeit ist der Gedanke, ohne Geld zu leben, eher eine Utopie, aber machbar...

Warum aber brauchen wir nun doch eine neue Kryptowährung?

Beim Bitcoin hat sich gezeigt, dass die dezentrale Datenhaltung eine großartige Idee war. Allerdings werden dadurch alle Transaktionen offengelegt, und beim sogenannten Mining wird unglaublich viel Energie verschwendet. Zudem ist der Bitcoin zu einem Spekulationsobjekt geworden. Dadurch können nur die Reichen noch reicher werden.

Die US-amerikanische Bank JP Morgan hat einmal öffentlich gesagt, dass der Bitcoin nichts taugen würde oder so ähnlich. Daraufhin fiel der Marktwert, und am nächsten Tag kaufte JP Morgan einen großen Anteil an Bitcoin auf, wodurch der Preis wieder stieg und JP Morgan um einige Milliarden reicher wurde. Zumindest wurde mir das so berichtet.

# Spekulationen sollten verhindert werden

Ich habe einmal mit Bernd Hückstädt, dem Erfinder des Gradido, an einem gemeinsamen Projekt gearbeitet. Wir wollten gemeinsam eine App für Gradido entwickeln. Der Gradido war eine großartige Idee. Shift ähnelt ihm sehr.

Leider konnte ich Bernd nicht davon überzeugen, dass wir unbedingt eine auf Zeit basierende Währung wie die Timebanks benötigen, um den Wert stabil zu halten. Da ich jedoch beharrlich darauf bestand, dass DIE Währung, die uns aus der Misere retten könnte, auf Zeit basieren muss, gingen wir wieder getrennte Wege. Nicht umsonst habe ich von Don Norman den Auftrag für eine auf Zeit basierende App erhalten. Das Universum weiß, was gut und richtig ist.

In Shift wird pro Stunde und Minute abgerechnet, um Fairness zu gewährleisten. Ein Softwareentwickler in Norwegen verdient also genauso viel wie ein Friseur in Indien, nämlich 60 LMC pro Stunde. Natürlich gibt es Unterschiede, je nachdem, ob ein Geselle, jemand mit 6 Semestern Studium und 20 Jahren Berufserfahrung oder ein Schüler seine Arbeitsstunde abrechnet.

Da die App weltweit eingesetzt wird, gibt es niemanden mehr, der beim Geldwechsel

Gebühren erhebt, und es gibt auch keine Billiglohnländer mehr. Die Waren werden also wieder lokal hergestellt, da es sich nicht lohnt, sie im Ausland herzustellen, da es dort genauso teuer ist, aber zusätzlich noch hohe Transportkosten anfallen.

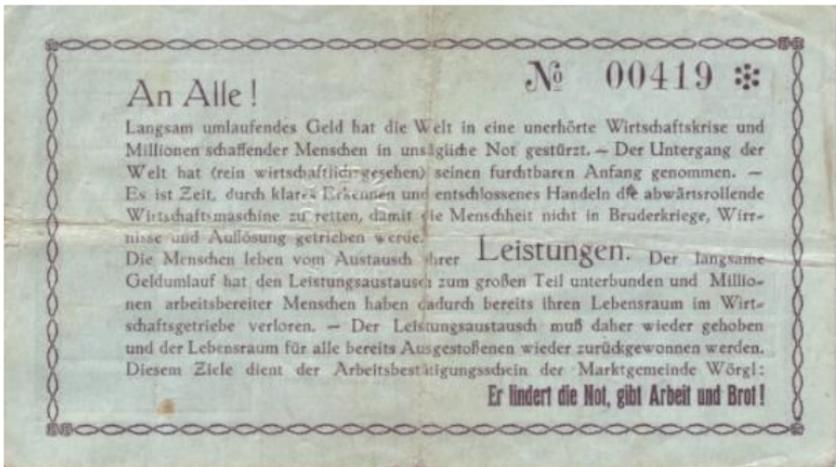
## **Das Horten sollte verhindert werden**

Derzeit haben wir das Problem, dass Menschen und Unternehmen viel Geld horten und es nicht wieder in Umlauf bringen. Dadurch wird dem Markt einerseits das Austauschmittel entzogen und andererseits wird das Geld nicht verwendet, um andere Waren zu kaufen und so die Wirtschaft anzukurbeln.

Deshalb nennen wir unsere Zahlungseinheit nicht Geld, sondern Liquid. Es soll eine Flüssigkeit sein, die die Wirtschaft in Bewegung hält.

Silvio Gesell fand damals heraus, wie man dieses Problem lösen könnte, und in der österreichischen Stadt Wörgl wurde seine Idee während einer Rezession vom Bürgermeister umgesetzt. Man nannte es das "Wunder von Wörgl".

Der Trick bestand darin, dass der Bürgermeister Gutscheine druckte und sie im Austausch für Arbeit ausgab.



Diese Gutscheine wurden überall wie Geld akzeptiert. Sie hatten jedoch eine Besonderheit: Ihr Wert nahm im Laufe der Zeit ab. Dadurch wurde man dazu gezwungen, die Gutscheine wieder in Umlauf zu bringen. Wenn man sie behielt, verloren sie an Wert. Das belebte damals die Wirtschaft sehr gut, bis die Regierung in Wien dies verbot.

Der von uns geschaffene LMC (Liquid Micro Coin) ist ebenfalls ein Schwundgeld. Er verliert jeden Tag 0,27 Prozent seines Wertes, sodass er nach 7 Jahren völlig wertlos ist. Wir schöpfen also diese Flüssigkeit, und sie verdunstet innerhalb der sieben Jahre wieder. Genau wie in der Natur, wie Bernd Hückstädt wahrscheinlich geschrieben hätte.

Wir nennen dies Demurrage oder auch Verfall.

## **Bezahlen ohne Internet**

Ich habe eine Zeit lang in den Bergen Portugals gelebt und hatte dort keinen Internetempfang. Ärmere Menschen haben oft generell keinen Zugang zum Internet.

Aus diesem Grund haben wir uns dafür entschieden, Zahlungen ohne Internet, NFC oder Bluetooth zu ermöglichen.

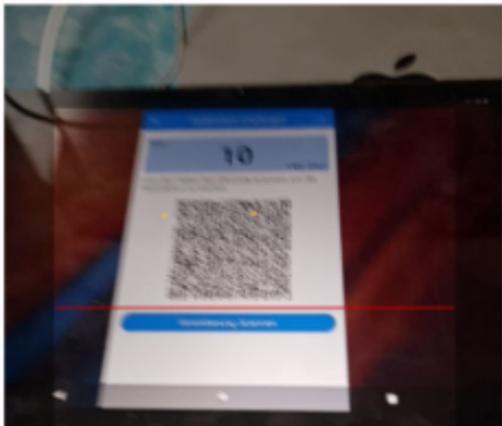
Ein anderer Grund ist folgender: Zunächst haben wir die P2P-Bibliothek IPv8 einer Universität in Delft, Niederlande, genutzt, um Transaktionen abzuwickeln. Dann wurde uns jedoch mitgeteilt, dass der Quellcode zwar veröffentlicht wurde, aber keine Lizenz vergeben wurde.

Also haben wir nach einer anderen Lösung gesucht. libp2p sah vielversprechend aus. Da jedoch bestimmte Funktionen in der Kotlin-

Version von libp2p noch nicht implementiert waren, konzentrierten wir uns auf die Go-Variante. Später haben wir die gesamte Geschäftslogik in Go geschrieben, um unter anderem das Auffinden von API-Schlüsseln im APK zu erschweren. In Kotlin könnte man mit einem Knopfdruck einen Teil des Codes rekompilieren und dann alle geheimen Schlüssel einsehen, was quasi eine Einladung für Hacker wäre. In Go wird Maschinen-Code erzeugt, was eine Rekompilierung zwar nicht verhindert, aber erheblich erschwert.

Da ich bereits einige Jahre zuvor eine technische Lösung für den Wertetransfer entwickelt hatte - wie gesagt, ich war in den Bergen ohne Empfang -, haben wir auch auf den Einsatz von libp2p verzichtet, und die Transaktionen werden nun über QR-Codes und Kamera-Scans abgewickelt.

Scanne den QR-Code, um das Angebot vom Empfänger zu erhalten.



Da wir die Transaktionen nur auf den Handys speichern und keine dieser Daten ins Internet senden, kann uns niemand abhören oder das System abschalten. Es funktioniert also genauso anonym wie das gute alte Bargeld.

Ich glaube, das ist unser Alleinstellungsmerkmal und der Grund, warum wir noch eine Kryptowährung brauchen.



# 2030 - Ein kurze Geschichte

## In einer kleinen Gemeinde in Schleswig-Holstein

Dirk fährt an diesem Morgen mit dem Fahrrad zum Markt, um frisches Gemüse zu holen, da er heute Abend Besuch von seinem Singkreis bekommt.

Auf dem Weg dorthin trifft er auf Erika, die ihn fragt, ob sie ihren Bruder, der aus Polen zu Besuch bei ihr ist, mitbringen soll.

"Ja, na klar", sagt Dirk, "Wir freuen uns über jede Seele, die bei uns mitsingt"

Am Markt angekommen, wird er von einer Schar Kinder umringt, die sich sichtlich freuen, ihren Lehrer zu sehen. Er sagt ihnen, das er ihnen morgen, also am Montag eine interessante Geschichte erzählen wird.

Nach dem er seinen Korb voll mit Gemüse gemacht hat, ruft ihm Angela hinterher. "Dirk, hier habe ich noch eine wunderschöne Birne für Dich. Lass sie Dir schmecken." Er bedankt sich bei allen Menschen auf dem Markt für dieses wundervolle naturbelassene Gemüse und witzelt.

"Könnt ihr euch noch an damals erinnern, wo es auf dem Markt nur noch behandeltes Gemüse gab und wir dafür auch noch Geld bezahlen mussten?" "Och hör bloss auf mit dem Schiet", rief ihm Hannes zu, "daran wolln mer uns gar nicht erinnern."

Zuhause angekommen, legte er das Gemüse in den Vorratskeller, in dem es trotz des heißen Sommers sehr kühl war.

Am Abend kamen dann ein paar Leute zu ihm um gemeinsam zu musizieren. Auch Vlado, der Bruder von Erika war da und hat einen Korb mit Früchten und seine Gitarre mitgebracht. Vlado wurde eingeladen, etwas auf seiner Gitarre zu spielen.

Er spielte einen alten Rainbow-Song, der in etwa so ging:

*I find my joy in the simple things, coming from the earth.*

*I find my joy in the sun that shines and the water that sings to me.*

*Listen to the wind and listen to the water, hear what they say.*

*Singing heya heya heya heya heya heya hey.*

Alle sangen mit und hatten Tränen der Freude in ihren Augen.

Während ein paar Menschen in die Aussenküche gingen, um das Essen zuzubereiten, debatierten ein paar der anderen über den Bau des neuen Blockhauses, das zu Ehren der Hochzeit von Jens und Jaquelin gebaut werden soll. Peter meinte, das er noch etwa 5 Bäume dazustiften könne, da er den Wald ausdünnen muss. Vlado, meinte, da er noch ein paar Wochen bleiben würde, er gerne beim Bau helfen würde, wenn es in Ordnung wäre. Auch Sylvia bot an, bei der Innendekoration helfen zu wollen. Bernd wurde gefragt, ob er seinen Bagger schon fertig repariert hat. Bernd meinte, dass er nur noch etwas Rapsöl benötigt, um eine Probefahrt machen zu können.

Gisella warf ein, das sie noch 20 Liter Rapsöl über habe und auch Jenny möchte gerne 10 Liter beisteuern.

"Hurra", sagt Dirk, "dann haben wir ja beinahe alles zusammen für das neue Haus und anbauen können wir dann später immer noch, wenn sich Nachwuchs eingestellt hat."

Jens und Jaquelin schauen sich verliebt in die Augen und küssen sich.

# Am Montag in der Schule

Dirk fuhr diesen morgen etwas früher in die Schule, denn er wollte Sandra eine Überraschung präsentieren. Sandra ist seine große Liebe, die im Büro der Schule arbeitet.

"Guten Morgen", rief er ihr schon von Weiten zu und umarmte sie dann zärtlich. "Was machst Du denn schon so früh hier?", wollte Sandra wissen. "Ich wollte Dich einfach noch mal umarmen bevor der Unterricht anfängt. Ach ja", sagte Dirk, "und dies hier wollte ich Dir geben." Er überreichte ihr ein kleines Paket, welches sie voller Vorfreude auspackte.

Es war eine Einladung zum Gitarrenkonzert von Guiseppe Pardonni, der aus Italien angereist war und eines der Lieblingsmusiker von Sandra ist. "Wie hast Du denn das schon wieder geschafft?", fragte Sandra.

"Nun, Du weißt doch, das ich auch Telepathie unterrichte und da habe ich einfach..."  
"Schon gut schon gut", warf Sandra ein. "Mmmh, ich liebe Dich du Tele-Spinner", und küßte ihn ganz leidenschaftlich auf den Mund.

Die Klasse, die Dirk unterrichtet ist nicht so eine herkömmliche Klasse, wie Du sie eventuell kennst, es war eher eine Mischung aus Kindern

im Alter von 6 - 12 Jahren und auch einige Jugendliche sind dort vertreten.

Dirk fing an, seine Klasse zu moderieren.

"Wer von euch möchte heute anfangen, eine Geschichte zu erzählen?", sagte Dirk.

Der kleine Peter hob die Hand und sagte, "Darf ich bitte."

Da sich sonst keiner meldete, gab Dirk den Redestab, der eigentlich kein Stab sondern eher eine kleine Figur darstellte, an Peter weiter.

Peter erzählte, wie er am Sonntag einen Fisch mit seiner Angel gefangen hat und seine Mutter ihn danach gleich ausgenommen und gebraten hat.

Peter gab den Stab nach Link in dem Kreis weiter.

Nachdem der Redestab eine komplette Runde gemacht hat und jeder von seinem Wochenende berichten konnte, kam der Stab zu Dirk zuzück.

"Heute möchte ich euch eine alte Geschichte erzählen, die sich vor einigen Jahren zugetragen hat. Es geht heute um das Geld." "Was ist denn Geld?", wollte die kleine Susanne wissen.

"Das war so ein Stück Papier mit einer Zahl und so einem Bild darauf.", erklärte Rüdiger, einer der Jugendlichen.

"Wozu brauchte man denn sowas?", wollte Tina wissen.

"Nun lasst es mich doch bitte einmal ausreden"

"Dann setze man sich in einen Bus und fuhr in die Stadt, betrat eines dieser gruseligen Gebäude, man kann diese Gebäude übrigens immer noch in dem Museum "Berlin" sehen. Da machen wir ja eventuell mal einen Ausflug hin, und sass dort den ganzen Tag und ordnete Papier mit Buchstaben und Zahlen darauf, gab dem Papier einen Stempel und legte es auf einen Stapel."

"Andere Menschen bauten den ganzen Tag Autos zusammen oder reparierten Fahrräder oder so was in der Art."

"Aber Du reparierst doch auch Fahrräder", warf Iris ein. "Ja", sagte Dirk, "das nenne ich aber nicht Arbeit, das mache ich doch nur wenn mal was kaputt sein sollte und das passiert ja nun wirklich nicht oft."

"Heute wird nichts mehr gebaut, was schon nach einem Jahr wieder kaputt geht. Das haben die Menschen damals nur so gemacht, weil sie Geld verdienen mußten. Da wurden die Sachen extra so gebaut, das sie kaputt gehen, damit man neue Sachen kaufen mußte."

"Ist das nicht ein bisschen dumm?, kicherte Bernd.

"Ja, aber wir haben ja daraus gelernt und das Geld abgeschafft.", erklärte Dirk.

"Nun bin ich aber mal gespannt.", sagte einer der Jugendlichen, obwohl er genau wußte, was jetzt

kommt, denn die Geschichte hat er schon ein paar mal gehört.

"Es war das Jahr 2023. Man munkelt, das ein Mann aus Berlin diese revolutionäre Idee hatte. Er entwickelte eine App, mit der jeder Mensch seine eigene Währung schöpfen und damit bezahlen konnte. Diese Währung nannten er LMC (Liquid Micro Coins)."

Die Kinder sahen sich verwirrt an.

"Stellt euch vor", fuhr Dirk fort, "dank dieser App konnten die Menschen plötzlich ihre eigenen Bedürfnisse befriedigen, indem sie Güter und Dienstleistungen direkt untereinander austauschten, ohne auf traditionelle Währungen angewiesen zu sein. Die LMC waren digital und konnten ganz einfach übertragen werden."

Die Schüler fingen an, die Bedeutung dieser Veränderung zu verstehen.

"Mit der Zeit erkannten die Menschen, dass es nicht darum geht, immer mehr zu besitzen, sondern die Ressourcen unseres Planeten nachhaltig zu nutzen und für das Wohl aller zu sorgen", erklärte Dirk. "Die Gemeinschaften begannen, sich zu vernetzen und sich gegenseitig zu unterstützen. Es entstand ein neues Verständnis von Gemeinschaft und Solidarität."

Die Kinder waren fasziniert von dieser Vorstellung.

"Niemand war mehr gezwungen, den ganzen Tag so eine Arbeit verrichten zu müssen, nur um die Miete für eine Wohnung zahlen, ein Auto abzustottern und um für Essen Geld ausgeben zu können."

"Hat das Essen, das hier einfach so wächst Geld gekostet?", fragte Tina.

"Was ist denn eine Miete und wozu brauchten die Menschen ein Auto?, wollte Ernst wissen.

"Ja", sagte Dirk, "die Menschen in der Stadt zumindest wußten nicht, das man Essen auch selber anbauen kann und ausserdem hatten sie keine Zeit dafür, es zu sähen und zu ernten, denn sie haben ja den ganzen Tag gearbeitet."

"Miete war etwas, unter der die Menschen lange gelitten haben. Sie mußten Geld dafür bezahlen, das sie ihr Bett in so einem Betonklotz aufstellen durften. Jeden Monat immer wieder. Es war damals noch verboten im Wald zu wohnen."

"Die Autos waren wichtig, weil die Menschen ja in die Stadt fahren mußten. Ausserdem waren diese armen Menschen so krank von der vielen Arbeit, das sie weder zu Fuß gehen noch mit dem Fahrrad fahren konnten, weil die Menschen noch sehr schwach waren, da sie kein gesundes Essen hatten."

"Was ist denn das mit dieser App genau?", fragte Bärbel.

"Ja, mit der App konnten die Menschen Dienstleistungen und Waren wie zum Beispiel, Reparaturen, Brot, Massagen, Gemüse usw bezahlen. Die Menschen waren also in der Lage, gesundes Essen vom Bauer zu bekommen. Und sie konnten selber ihre Dienste anbieten und sich bezahlen lassen. Keiner mußte mehr Geld an den sogenannten Staat ausgeben, der dies dann benutzt hatte um Kriege zu finanzieren, aber das erzähle ich euch ein anderes Mal, wenn ihr etwas älter seid."

"Da es den Menschen dann wieder gesundheitlich besser ging, haben sie ihre Autos verkauft oder nach Afrika verschenkt, haben ihre Fernseher weggeschmissen, weil sie diese gar nicht mehr brauchten, haben angefangen sinnvolle nachhaltige Produkte zu erstellen, die nicht mehr kaputt gingen und so weiter und so fort."

"Da die Menschen nicht mehr arbeiten wollten, nur um ihre Miete zahlen zu können, sind sie aufs Land gezogen und haben dort angefangen kleine Siedlungen zu bauen. Erst einfache Jurten, dann Tiny-Häuser und Blockhäuser. Diese haben sie aus Material gebaut, was sie günstig bekommen haben, weil Firmen es als Müll wegwarfen."

"Naja, und Holz für diese Gebäude wächst kostenlos im Wald."

"Die Menschen mußten keine Miete mehr zahlen. Das Essen wuchs kostenlos im Garten. Autos mußten sie auch nicht mehr bezahlen, also konnten sie auch aufhören zu Arbeiten."

"Die Tiere arbeiten ja auch nicht.", sagte Peter.

"Genau", sagte Dirk, "die Tiere waren damals cleverer als die Menschen. Obwohl die Menschen dachten, sie wären clever.", schmunzelte Dirk.

"Und wo ist die App jetzt.", wollte Iris wissen.

"Ich habe noch so ein altes Handy, da ist die App noch drauf. Ich benutze sie aber nicht mehr. Keiner hier benutzt noch Geld oder ne App. Wozu auch."

"Die Menschen begannen, nachhaltige Lebensweisen zu entwickeln. Sie bauten ökologische Häuser, nutzten erneuerbare Energien und betrieben ökologische Landwirtschaft. Sie erkannten, dass sie Teil eines größeren Ganzen sind und dass ihr Handeln Auswirkungen auf die Natur hat."

Dirk lächelte, als er die Begeisterung in den Augen seiner Schüler sah.

"In dieser neuen Welt gab es keine verschmutzte Luft mehr, keine ausgedehnten Monokulturen und keinen Müll, der die Ozeane verschmutzte. Die Menschen lernten, im Einklang mit der Natur zu leben und sie zu schätzen. Sie schützten die Wälder, reinigten die Gewässer und achteten darauf, dass jede Handlung nachhaltig war."

"Und das Beste daran war", fuhr Dirk fort, "dass die Menschen nicht mehr gezwungen waren, den Großteil ihres Lebens damit zu verbringen, für Geld zu arbeiten. Sie hatten Zeit für ihre Familien, für kreative Projekte und für die Dinge, die ihnen wirklich am Herzen lagen. Sie lebten ein erfülltes Leben."

Die Klasse war erstaunt und inspiriert von dieser Geschichte aus der Vergangenheit, die eine Zukunft voller Hoffnung und Möglichkeiten beschrieb.

"Und nun, sind wir Teil dieser Entwicklung geworden", verkündete Dirk stolz. "Wir haben die Lektionen aus der Vergangenheit gelernt und eine Welt geschaffen, in der jeder Mensch sein volles Potenzial entfalten kann und in Harmonie mit der Natur lebt. Unsere Gemeinschaft ist stark und solidarisch, und wir teilen die Ressourcen, die uns die Erde bietet, gerecht."

Die Schüler lächelten, denn sie wussten, dass sie die Gestalter dieser Zukunft waren.

"Kinder, lasst uns gemeinsam dies weiter vorantreiben. Lasst uns unsere Talente und unsere Kreativität nutzen, um eine Welt zu erschaffen, in der Frieden, Gerechtigkeit und Nachhaltigkeit herrschen. Wir sind die Generation, die den Unterschied machen kann."

Und so endete Dirk seine Geschichte, während die Schüler voller Vorfreude und Tatendrang in die Zukunft blickten. Sie hatten verstanden, dass sie die Macht hatten, eine Welt zu formen, in der Wirtschaft, Regierungen und Umweltschutz im Einklang standen. Gemeinsam würden sie diese Utopie Wirklichkeit werden lassen.

# Abends Zuhause

Als Tina Abends mit ihren Eltern am Essen ist, erzählte sie von Dirk's Geschichte und fragte ihre Eltern, wie sie das damals erlebt hatten.

Tinas Mutter fing an und meinte, "Das ist Dirk seine Lieblingsgeschichte, ich höre sie auch ganz gerne und erinnere mich daran gerne zurück." "Da ich damals meinen Beruf als Pflegerin nicht mehr ausüben durfte, weil ich mich geweigert hatte bei den Versuchen der Pfarmaindustrie mitzumachen, hatte ich nur noch ganz wenig Geld und konnte mir weder das Kino, noch das Theater und schon gar keine Restaurants leisten." Tina schaute ihre Mutter traurig an. "Ich habe dann auch von diesem neuen Geld gehört, welches wir uns selber schöpfen konnten. Und habe angefangen Yoga-Unterricht zu geben und bekam LMC als Dank von meinen Teilnehmern. Bei einem Kurs hab ich dann auch Deinen Vater kennengelernt.", sagte Tinas Mutter und warf Tinas Vater einen verliebten Blick zu.

"Und Du Papi, was hast Du so erlebt zu dieser Zeit?", wollte Tina wissen.

Tinas Vater hohlte tief Luft und fing an zu erzählen, "Das ist ja nun schon etwas länger her. Und ja, damals habe ich Deine Mutti beim Yoga kennengelernt. Da auch ich damals nach einem

Burnout aufhören mußte zu arbeiten, hatte ich auch kein Geld über und war froh, das ich die Yoga-Stunden mit LMC bezahlen konnte. Ich habe vorher in einer Bank gearbeitet und war es leid, den Menschen ihre Träume von einem Eigenheim zu zerstören, weil sich die Leute damals zwar die Zinsen leisten konnten, ich sie aber darauf aufmerksam machen mußte, das sich die Zinsen von heute auf morgen verdoppeln, wenn nicht sogar verdreifachen konnten und sie dann bestimmte Schwierigkeiten bekommen, die Hypothek zurück zahlen zu können und dann nicht nur ihr Haus sondern auch ihre Ersparnisse wieder verlieren würden. Wie gesagt ich war deshalb ausgebrannt und konnte nicht mehr mit arbeiten." Tinas Vater hatte Tränen in den Augen und erzählte weiter, "Ich wußte damals gar nicht was ich mit all meiner Zeit anfangen sollte und fang an Alkohol zu trinken um meinen Schmerz nicht so stark spüren zu müssen. Ab und zu dachte ich darüber nach, was ich Gutes in die Welt bringen könnte, aber als Bankkaufmann hatte ich ja nicht mal ein Handwerk gelernt. Bis ich eines Tages von einem lieben Freund erfahren habe, was damals in Berlin gestartet war. Da gab es tatsächlich eine App mit der wir die Welt verändern könnten." "Davon hat Dirk uns heute in der Schule erzählt", strahlte Tina.

"Ja", sagte Tinas Vater, "so eine simple Idee und soo große Ausmaße. Da ich mich als Banker ja mit Geld auskannte und auch wußte, was die

Banken weltweit falsch machen, habe ich mich ins Zeug gelegt und habe mir Gedanken gemacht, wie wir die Menschen davon überzeugen können, dieses neue Geld auch als Zahlungsmittel zu nutzen." "Ich kannte ja all unsere Bankkunden persönlich und habe dann das Gespräch mit ihnen gesucht. Ich hatte wieder eine Aufgabe und hatte zu der Zeit ganz vergessen, Bier zu trinken.", lachte Tinas Vater, während ihm die Tränen in die Augen schossen. Tinas Mutter stand auf und umarmte ihren Mann ganz liebevoll, um ihn zu trösten.

"Och Papi, so habe ich Dich ja lange nicht mehr gesehen. Ich hab Dich ganz doll lieb.", sagte Tina und umarmte ihren Vater auch.

Nach einer Weile fing Tinas Vater an zu ergänzen, "Ich habe den Menschen einfach die Wahrheit erzählt, wie die Großbanken Geld aus dünner Luft schöpfen, jedes Mal wenn sie einen Kredit vergeben. Da das Geld aus dünner Luft geschöpft wird, hat es gar keinen Wert. Nicht so, wie das viele Jahre zuvor Gold-gedeckte Geld. Und da das Geld keinen Wert hatte, war es sogar nachhaltiger dieses neue Geld zu nutzen, weil man damit etwas Gutes für seine Gemeinschaft getan hatte. Man half seinen Mitmenschen wieder am gesellschaftlichen Leben teilnehmen zu können."

Tinas Mutter fiel ein, "Ach Du warst das damals, der mir von Shift erzählt hatte. Jetzt fällt es mir

wieder ein. Da hast Du noch eine Krawatte  
getragen."  
Tinas Mutter mußte grinsen. Und alle lachten.

# Abends bei Peter

Auch Peter erzählte am Abend von Dirk's Geschichte und fragte seine Eltern, wie sie das damals erlebt hatten. Peters Vater fing an, "Boah, ich war damals so skeptisch, schon wieder so ein neuer Krypto-Mist. Wieviel wollen die davon denn noch rausbringen. Wer braucht denn sowas überhaupt, wir haben doch den stabilen Euro. Nur der ist das wert, was draufsteht. Ist das überhaupt legal?" "Aber Du hast Dich doch fleissig dafür eingesetzt, das dies nun unser neues legales Zahlungsmittel wird.", sagte Peters Mutter. "Ja", antwortete Peters Vater, "das war zu der Zeit, als wir beide uns kennengelernt hatten, aber vorher war ich stark dagegen. Das wurde in Wörgl ja auch verboten worden von der Regierung. Und Silvio Gesell, der das damals erfunden hatte, war eine Rechter." "Aber wir wissen doch alle, warum Silvio Gesell von den Regierungen in die rechte Schublade gesteckt wurde.", warf Peters Mutter ein. "Ja", stellte Peters Vater klar, "ich weiß ja. Mir wurden zu der Zeit ja auch die Augen geöffnet, das diejenigen, die das meiste Geld hatten, mit ihrem vielen Geld Leute bestochen haben, damit sie ihre Meinung ein wenig abändern und dann andere Menschen, die sie nie kennengelernt haben, öffentlich zu diffarmieren. Die wollten ja nicht, das ihr tolles Geldsystem, welches sie so reich gemacht hat, abgeschafft wird." "Glücklicher Weise schreiben

wir heute unsere Geschichtsbücher selber. Damals war es noch History, was aus dem Englischen kommt und "seine Geschichte" bedeutet."

"Wo hast denn dit her Harry?", wollte Peters Mutter wissen und schmunzelte.

Peters Mutter erzählt, "Damals habe ich noch als Lehrerin an einer Realschule gearbeitet. Dort haben wir die Kinder streng dazu erzogen still zu sein und nur zu reden, wenn die Erwachsenen sie dazu auffordern. Auch war es unser Ziel, jedem Kind den Weg in den Arbeitsprozess zu ermöglichen."

"Das haben damals alle so gemacht und mir wurde es so beigebracht. Uns war nicht bewusst, dass wir da gehorsame Sklaven züchten, um es mal beim Namen zu nennen. Das wurde uns damals aber mehr und mehr bewusst. Die Politiker redeten ständig von Vollbeschäftigung, während mehr und mehr Maschinen und Computer die Arbeit für uns verrichtet haben und es durch die künstliche Intelligenz dann noch schlimmer zu werden drohte."

"Zum Glück sind wir dann irgendwann aufgewacht und die Regierung hat dann einfach nur noch die Produkte besteuert, uns aber keine Lohnsteuer mehr abgenommen. All die Reichen, die teure Autos fahren wollten oder riesige Villen bezogen, haben dann die Steuern ganz alleine

zahlen müssen. Ob sie es wollten oder nicht. Normale Menschen wie wir brauchten keine Autos oder Villen."

"Konsumsteuer heißt es heute noch. Ich glaube es war Götz Werner, der dies Modell im Zusammenhang mit der Finanzierbarkeit vom Bedingungslosen Grundeinkommen, erfunden hatte. Es wurde einfach die Lohnsteuer weggelassen und statt der Mehrwertsteuer kam eine Konsumsteuer in Höhe von 100% auf jede Ware. Dadurch zahlten quasi auch die Maschinen eine Lohnsteuer und es war genug Geld für das BGE vorhanden. In Dänemark wurde dies schon viel früher gemacht. Dort sieht man auch viele Oldtimer, weil es sich dort gelohnt hat, diese zu restaurieren anstatt sich neue Autos zu kaufen."

# Erik



An Erik haben die Entwickler von Shift leider nicht gedacht, denn Erik war nicht nur obdachlos sondern auch noch bettelarm und konnte sich natürlich kein Handy leisten um Liquid schöpfen zu können. Aber so schlimm ist das ja nun auch wieder nicht, denn nun hatten die Menschen zum einen viele Euros über, die sie Menschen in Not wie Erik geben konnten und da es in dem kleinen Dorf in Schleswig-Holstein genug zu Essen gab, haben sie Erik immer an ihrem Tisch mitessen lassen und sie haben alle zusammen angepackt, um Erik eine kleine Hütte zu bauen. Erik ist heute der Bürgermeister dieses wundervollen Dorfes im Norden von Deutschland.



# Architektur der App

## Front End

Das Front End, also die App selber wurde in Kotlin und Jetpack Compose geschrieben. Mit Compose geht Google einen neuen Weg, der sich so ähnlich anfühlt wie Flutter, weg vom umständlichen XML-Ansatz.

Eigentlich sollte mir der XML-Ansatz gefallen, da ich selber einmal, noch bevor XAML (Microsoft) und QML (Qt) heraus kamen, einen ähnlichen Weg gegangen bin, um Java-Swing-Anwendungen zu serialisieren. Ich fand das elegant, weil es in HTML ja ähnlich funktioniert.

Wenn ich nun aber die Integration von Kotlin im Zusammenhang mit Compose und der Möglichkeit für den Preview sehe, macht dieser Ansatz viel mehr Spaß. Apropos eine Funktion in Kotlin heißt: **fun = Spaß**

Ich setze einfach mal voraus, das Du in der Lage bist, dir selber Kotlin und Jetpack beizubringen. Zum einen gibt es darüber genug Videos, Lehrbücher und Tutorial und zum anderen habe ich es auch selber grad erst mit Hilfe von ChatGPT gelernt. Ja richtig, mit Hilfe der KI.

ChatGPT hat auch die ganze Übersetzung ins Französisch, Spanisch, Portugiesisch und ins Esperanto übernommen.

Ich habe einfach einen Englischen Text, wie zum Beispiel eine Zeile aus der *values/strings/strings.xml* Datei genommen

```
<string name="save_changes">Save  
changes</string>
```

und ChatGPT gebeten

```
Please translate to German, Portugues,  
Spanish, French and Esperanto
```

Dann habe ich die Deutsche Version überprüft, ob es sinngemäß übersetzt wurde und dann alle Übersetzungen in die jeweilige *strings.xml* Datei kopiert.

Auch kannst Du Code in einer Programmiersprache Deiner Wahl schreiben und ChatGPT bitten, es nach Kotlin zu übersetzen. Wenn es Java-Code ist, dann macht Android Studio das bereits für Dich, in dem du den Code einfach in eine Kotlin (kt) Datei reinkopierst.

Lediglich die Unit-Tests solltest Du selber schreiben, um zu überprüfen, das der Code, der von der KI kommt genau so funktioniert, wie Du es möchtest.

Hier habe ich in der Go Version schon fast TDD (Test Driven Development) gearbeitet. Dazu aber später mehr.

## Aufbau

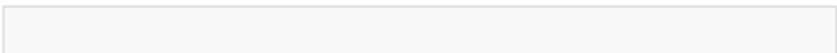
Wie in dem AndroidManifest.xml beschrieben, wird die MainActivity geladen, welche derzeit die einzige Activity ist.

Dort initialisiere ich den barcodeView der leider noch nicht als Composable vor lag.

Ich hatte zwar schon den BarCodeScanner aus dem ML-Kit von Google genutzt, aber was ich da gesehen habe, hat mich abgeschreckt. Sobald ich die Kamera eingeschaltet hatte, habe ich im LogCat Nachrichten wiedergefunden in dem kontinuierlich Daten gesendet wurden. Zumindes stand im Log sowas wie "sending data" und da habe ich 1 und 1 zusammengezählt und das ML-Kit (MaschineLearning) wieder rausgeschmissen. Nenn mich paranoid, aber ich mag den Gedanken nicht, das Google Daten sendet, während ich die Kamera auf habe.

Wie dem auch sei, dann machen wir das halt OldSchool mit nem AndroidView.

Des Weiteren wird der LocalManager initialisiert, der für die Übersetzung der Sprachen zuständig ist, bzw. je nach gewählter Sprache, den dementsprechender Text liefert.



```
LocaleManager.init(applicationContext,  
resources)
```

Der LocalManager benötigt den Context für den PersistenceManager mit dem User-Settings geladen werden und resources wird für die übersetzten Texte benutzt.

Hier das init() des LocaleManager.

```
1 fun init(context: Context, resources:  
Resources){  
2     languages.clear()  
3  
languages.add(resources.getString(R.string.  
4  
languages.add(resources.getString(R.string.  
5  
languages.add(resources.getString(R.string.  
6  
languages.add(resources.getString(R.string.  
7  
languages.add(resources.getString(R.string.  
8  
languages.add(resources.getString(R.string.
```

Mit dem init() der Go Lib initialisieren wir diese Bibliothek, die den Pfad für Dateien benötigt.

```
init(applicationContext.filesDir.absolutePa
```

Dann teste ich, ob der User schon registriert ist, in dem geprüft wird, ob die Datei shift.db schon existiert.

```
val hasJoined = remember {  
    mutableStateOf(hasJoined()) }  
}
```

Und ich prüfe, ob der User die Warnung am Anfang gelesen hat.

```
val hasSeenDeleteWarning = remember {  
    mutableStateOf(false) }  
}
```

Nicht das ein User einfach die App deinstalliert und sich wundert, das sein Konto gelöscht wurde.

Danach wird ein Array mit NavigationItem befüllt, welche im DrawerView angezeigt werden.

```
1 val list = mutableListOf(  
2     NavigationItem("home",  
Icons.Default.Home,  
stringResource(R.string.navigation_home)),  
3     NavigationItem("scooping",  
Icons.Default.AttachMoney,  
stringResource(R.string.scooping_menuitem))  
)
```

Die Plugins werden geladen, welche sich in die Liste einreihen können

```
PluginManager.loadPlugins(LocalContext.current)
list)
```

und zusätzliche Seiten in der Liste aufgenommen, welche zwar für die Navigation dienen, aber nicht im Menü angezeigt werden.

```
1
list.add(NavigationItem(id="receive_gratitu
2
list.add(NavigationItem(id="receive_gratitu
```

Je nach State wird nun die Intro-Seite, die Join-Seite oder die Hauptseite, die einfach nur eine Webseite anzeigt, geladen.

In der Webseite können wir vom Webserver aus, Neuigkeiten verbreiten, wenn es zum Beispiel ein Update geben sollte. Auch hat diese Seite eine statische Version die angezeigt wird, wenn der User offline ist (siehe im `assets` Verzeichnis). Der Rest der App ist nativer Kotlin Compose Code, der eigentlich selbsterklärend sein sollte, wenn man sich ein wenig mit Compose auskennt.

Für Anfänger mag es etwas merkwürdig aussehen, wenn man solchen Zeilen wie diesen über den Weg läuft,

```
1 var code by remember {
mutableStateOf("") }
```

```
2 var showScanner by remember {  
mutableStateOf(true) }
```

was mich mal kurz wieder von Compose entfernt hat, aber da ich das mit dem Preview so cool fand, habe ich es mir dann doch noch mal angetan, es zu versuchen, es zu begreifen.

```
1 @Preview(showSystemUi = true)  
2 @Composable  
3 fun JoinFormPreview()  
4 {  
5     val hasJoined = remember {  
mutableStateOf(false) }  
6     JoinForm(hasJoined, "de")  
7 }
```

Nachdem ich schon fast die komplette Business-Logik in Kotlin fertig hatte, habe ich mich daran erinnert, das ja der komplette Code aus Byte Code rückkompiliert werden kann. Man konnte also Passwörter und API-Schlüssel im Quelltext einsehen mit nem einfachen Doppelklick in Android Studio. Aus diesem Grund habe ich die Logik dann noch einmal komplett neu in Go geschrieben. Denn Go erzeugt nativen Maschinencode, was es den Hackern etwas erschwert, selbigen Code zu lesen.

## Storj und Public Keys

In der Applikation kann man bereits jetzt schon Freunde hinzufügen, in dem diese ihren Public Crypto Schlüssel und einen Storj-Access-Token via QR-Code austauschen.

Für die Chat-Funktion und später für ein Micro-Blogging-Plugin, haben wir uns entschieden, Storj zum Speichern der Nachrichten zu nutzen. Storj bietet jedem Nutzer 25 GB freien Speicherplatz an, die Daten werden auf DHT-Nodes verteilt und sollte für unsere Zwecke sicher genug sein, da alle Nachrichten zusätzlich von Go Lib verschlüsselt werden. Storj kümmert sich um die Verschlüsselung beim Transport.

Storj ist ausfallsicher, da die Daten auf mehrere Nodes verteilt werden.

Die Chat-Nachrichten liegen dort nur temporär, und werden dann lokal gespeichert.

Die angedachte Blogging-Funktion a la Facebook oder ein Marktplatz auf dem man Waren anbieten kann, die in LMC gehandelt werden, speichern die Daten dann für längere Zeit.

Jeder User hat dann eine Wall a la Facebook und einen Marktplatz a la Amazon auf seinem Storj-Storage. Und nur deren Freunde können diese Daten einsehen. Eventuell noch die Freunde von Freunden, indem die Daten bei einem Freund lokal gespeichert werden.

Und nein, Plugins werden weder den Public Key noch den Storj-Access-Token auslesen dürfen. Aus diesem Grund regelt die Go Lib den Transport der Daten.

## API

Um für die Plugins eine Schnittstelle bereit zu stellen, habe ich die shiftapi eingebaut. Die API kann von einem Plugin verwendet werden, um zum Beispiel Nachrichten von User zu User zu senden. Hier habe ich zum Beispiel das Theme reingesteckt, damit die Plugins das selber *Look and Feel* haben wie die Haupt-Applikation. Ein wichtiger Punkt, wenn man an Usability und Corporate Identity denkt.

Die Messaging Funktionen sind in der Go Lib implementiert, wo intern auf Storj zugegriffen wird um Nachrichten dezentral zu speichern. Als Beispiel dient hier das Chat-Plugin, an dem ich derzeit noch arbeite, dessen Code aber bereits verfügbar ist unter Plugins/SamplePlugin.

Dies Chat-Plugin dient nur als Muster und es kann gerne jemand anderes komplett neu schreiben, denn dazu fehlt mir grad die Zeit. Eventuell sollten wir Entwickler uns da vorher abstimmen, um Ressourcen zusammenlegen zu können oder um zu sehen, ob ein Plugin schon in Arbeit ist. Auch wenn es für die User ja schön sein kann, auszuwählen. Es würde Redundanzen erzeugen.

Solltest Du ein Feature Request an die API haben, sehe bitte von Push Requests ab, sondern teile uns deinen Request erst einmal per Email mit, bevor du anfängst etwas zu implementieren, was wir aus Sicherheitsgründen oder aus welchem Grund auch immer ablehnen könnten. Lass uns also bitte vorher abklären, ob wir dies in die App integrieren wollen.

Die App ist nicht umsonst so klein wie es geht gehalten, um unnötige Fehlerquellen wegen Komplexität auszuschließen.

Ja, ich habe mal für ne Bank gearbeitet und kenne deren Probleme.

## **Business Logik**

Wie schon erwähnt, wurde die Business-Logik in Go geschrieben um es Hackern zu erschweren, unsere App zu faken um finanzielle Vorteile zu erschleichen.

Ich persönlich fand dies aber als eine wunderbare Möglichkeit dafür zu sorgen, das die Lib zum einen TestDriven erstellt wurde und zum anderen ist dies ein interessanter Ansatz für Cross-Plattform, welchen ich auch schon in Büchern über Python, Qt und PySide6 genutzt habe. Nicht umsonst war ich mit meinen Python GUI Büchern in den TopTen auf Amazon unter Cross-Plattform.

Das hat sich unser Verein auf die Fahne geschrieben, das wir Software schreiben, welche unter Linux, Windows und Mac läuft, um den

Umstieg auf Linux zu erleichtern, weil unsere User dort eher ihre Privatphäre erhalten.

Der Kern der Go Lib ist die Kryptografie. So verwenden wir einen Algorithmus, der bei der Erstellung der Lib zufällige Zahlen erzeugt, aus denen dann ein Schlüssel generiert wird, mit dem alle genutzten Dateien verschlüsselt und entschlüsselt werden. So zum Beispiel der Account, der in der Datei shift.db gespeichert wird. Zusätzlich wird das Änderungsdatum der Datei selber in diesen Schlüssel eingewebt, so dass wir täglich einen neuen Schlüssel generieren.

Die API-Schlüssel für den Zugang auf den Webservice, der nur temporären Charakter hat, verschlüsseln wir bereits mit dem generierten Key so dass dieser API-Schlüssel auch nur verschlüsselt im Binary vorliegt.

Natürlich ist dies keine 100 prozentige Sicherheit, die gibt es in der IT nicht, aber es erschwert es möglichen Hackern enorm. Eigentlich würde ich mich ja freuen, wenn jemand unser Baby versucht zu hacken, denn dann sind wir auf dem richtigen Weg mit dieser App.

Des Weiteren hat die Lib eine nahezu 99 prozentige Testabdeckung, welche uns besser schlafen lässt. Das Testen erledigt sich fast von selbst mit den richtigen Visual Studio Code Plugin.

# WebService

Den Webservice wollen wir nur temporär nutzen.

Wir werden den Webservice in Phase III abschalten, um diese Abhängigkeit zu verlieren. Dann verbreitet sich die App ohne das Multilevel-Marketing von selber, denken wir. Dann ist die App wirklich dezentral und kann nicht mehr von Aussen gestört werden...

Nein, ich habe keine Paranoia, aber ich habe sehr früh, ich glaube in meinen 20ern Joseph Murphy gelesen (Die Macht des Unterbewusstseins), was mir sehr geholfen hat. Nein das ist nicht der Murphy, nach dem Murphies Gesetz benannt wurde, aber dies Buch hat mich hieran erinnert.

Murphy's Gesetz, auch bekannt als das Gesetz des unwahrscheinlichen Eintretens, geht auf den US-amerikanischen Ingenieur Edward A. Murphy Jr. zurück. Edward A. Murphy Jr. formulierte das Gesetz während eines Tests für die US Air Force in den späten 1940er Jahren. Murphy stellte fest, dass ein Schutzsystem, das er entwickelte, nicht wie erwartet funktionierte, und kommentierte:

"Wenn es zwei oder mehr Möglichkeiten gibt, etwas zu tun, und eine davon in einer Katastrophe endet, dann wird jemand es so machen."

- Edward A. Murphy Jr.

Dieser Ausspruch wurde später als Murphy's Gesetz bekannt und hat sich als populäre Maxime etabliert.

Die Nutzung des WebServices kann bereits jetzt schon in der App (bei der Registrierung) abgeschaltet werden, für den Fall das unser Webserver vom Netz genommen wird.

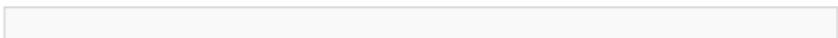
Das Funktioniert wie folgt.

Konnte der Einladungscode nicht vom WebServer verifiziert werden, weil entweder das Netzwerk nicht zur Verfügung steht oder aber der Webservice nicht mehr läuft, dann gibt es eine Entsprechende Fehlermeldung mit dem Hinweis das man auch im Offline-Modus weitermachen kann, wenn man den Button erneut drückt.

Im Offline-Modus kann man zwar keine Freunde mehr einladen und zusätzliche LMC schöpfen, aber dieser Modus schaltet den Zugriff auf den Webservice ab und ermöglicht die App auch offline zu laufen.

Derzeit kann man den Modus nur verlassen, in dem man die App deinstalliert und erneut installiert, was zur Folge hat, das das Konto zwischendurch gelöscht wird.

Der Webservice ist auch sehr übersichtlich. Er hat nur 4 sogenannte Routes. Ein hello\_world (ping)



```
1 @app.route('/')
2 def hello_world():
3     return 'Hello here is the
webservice of Shift!'
```

## ein register

```
1 @app.route('/register',
methods=['POST'])
2 def register():
3     content = request.json
```

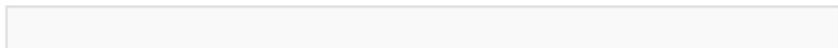
## ein setscooping

```
1 @app.route('/setscooping',
methods=['POST'])
2 def scooping():
3     content = request.json
```

## und ein matelist

```
1 @app.route('/matelist',
methods=['POST'])
2 def friendlist():
3     content = request.json
```

Die Datenbank dahinter ist auch sehr  
übersichtlich



```
1 CREATE TABLE account
2 (
3     uuid CHAR(52) NOT NULL PRIMARY KEY,
4     ruuid CHAR(52) NOT NULL,
5     name VARCHAR(250) NOT NULL,
6     scooping BIGINT NOT NULL,
7     country VARCHAR(30) NOT NULL,
8     language VARCHAR(10) NOT NULL,
9     FOREIGN KEY (ruuid) REFERENCES
account(uuid)
10 );
```

Country benutzen wir nicht mehr, um Länder zu speichern, sondern Kontinente.

Das soll eigentlich widerspiegeln, dass wir der Meinung sind, dass wir die Grenzen abschaffen sollten, damit die Freizügigkeit der Menschen überall gewährleistet wird.

Ja, das ist noch ein langer Weg bis dahin, aber wenn wir erst mal die Regierungen hinter uns lassen, dann brauchen wir auch keine Grenzen mehr. Ja, ich bin ein Träumer, aber was wäre die Welt ohne Träumer.

Imagine there's no heaven  
It's easy if you try  
No hell below us  
Above us only sky  
Imagine all the people  
Living for today... Ahhh..."

Imagine there's no countries  
It isn't hard to do  
Nothing to kill or die for  
And no religion too"

You may say I'm a dreamer  
But i'm not the only one  
Some day  
you might join us  
And the world will be as one  
-John Lennon



# Plugins

Hier ein paar Beispiele und Vorschläge was für Plugins, aus meiner Sicht Sinn machen könnten.

## Independent Musik

Diese Idee kommt zwar nicht von mir und das wird es auch bestimmt schon geben, aber halt noch nicht in Shift.

Stelle Dir vor Du bist Musiker und möchtest Deine Musikvideos oder Audiotracks online stellen. Statt es auf YouTube oder Vimeo oder so zu machen, wor Du Gefahr läufst, das es zensiert wird oder das Du es nicht monetarisieren kannst, weil ein anderer Musiker einen Song mit einer ähnlichen Melodie komponiert hat, oder warum auch immer, stellst Du es auf Storj dezentral zur Verfügung, und bindest es ins Plugin ein. Mit anderen Worten, die User speichern den Link zu ihrem Medium in der App und teilen es mit allen Usern und das Plugin zeigt all diese Videos an. Quasi eine Alternative zu YouTube. So mit ohne Zensur deren Marketing-Methoden.



Jeder, der deine Musik mag kann die Videos direkt in der App anschauen und Dir LMC spenden. Oder Du implementierst das Plugin so, dass der User jedes Mal, wenn er den Song hört, einen gewissen Mikrobetrag LMC an den Musiker sendet.

Ein LMC ist ja vergleichbar mit einer Minute Arbeit. Und wenn wir täglich 10 LMC schöpfen, dann können wir uns 10 Videos leisten. Man sollte den Betrag am besten nur einmalig zahlen müssen pro Video bemerke ich gerade.

# Paletten-Standorte

Neulich bin ich durch Berlin spaziert und sehe, das dort zwei Holzpaletten am Strassenrand stehen.

Ich mußte meine Geschrächspartnerin leider in ihrem Redefluss unterbrechen, weil ich das erst mal im Bild festhalten wollte.



Wie wäre es, wenn wir mit der App Standorte von Paletten melden würden, sie mit einem Geo-Tag versehen und andere Nutzer können dorthin mit

ihrem Lastenfahrzeug oder dem Auto fahren und die Palette abholen.

Ihr wisst ja, Holz ist kostbarer Rohstoff, den man nicht nur zum Heizen sondern auch zum Bauen von Hütten, Spielzeug usw. nutzen kann.

# Standorte leerer Häuser

Angenommen wir hätten Gesetze wie in Spanien, das man leere, verlassene Häuser einfach bewohnen dürfte.

Und dort solange wohnen könne, bis der Besitzer irgendwann seinen Anspruch anmeldet.

Ich denke dabei an die verlassenen Dörfer in Nord-Spanien. Dort sind die jungen Leute, die dort gewohnt haben in die Großstädte gezogen, um dort arbeiten zu können.

Deren Häuser stehen seit Jahren leer. Eventuell wohnen dort noch vereinzelt ältere Menschen, die in Madrid und Barcelona keine Arbeit finden würden und auch ihren Besitz nicht aufgeben wollen.

Stellt euch vor es gäbe Menschen, die sagen wir mal 5 Häuser besitzen und wegen Steuerhinterziehung und anderen Delikten, die man im allgemeinen Reichen Menschen zusprechen kann, im Gefängnis landen. Nun kann dieser Mensch, wenn er dann auch keine Kinder hat, die 5 Häuser gar nicht bewohnen.

Ich denke dabei gerade zu Bauchi, der eine seit Jahren verlassene Finca von einem Tennisstar besetzt hatte.

Oder denkt an den Lockdown. Alle mußten Zuhause bleiben, die vielen vielen Häuser stehen leer, während ärmere Menschen sich keine Wohnung leisten können und auf der Strasse leben müssen.

Hier in Berlin sind so viele leere Wohnungen, die von den Miethaien völlig überteuert angeboten werden, um den Wohnraum zu verknappen. Ich plapper da einfach mal etwas nach, was ich so gehört habe.

Aber Fakt ist ein Großteil der Berliner Wohnungen stehen leer, weil zu teuer, während andere Menschen auf der Strasse leben müssen.

Und nun hier die Idee.

Wenn wir nun eine leere Wohnung oder ein Haus finden, können wir das in der App eintragen, mit einem Geo-Tag versehen und Wohnungssuchende können dies auf einer Karte sehen und dort hingehen und einziehen.

Ja, ich weiß derzeit ist es noch illegal. Es gibt aber Beispiele wo dies bereits geklappt hat. Siehe zum Beispiel das [Gängeviertel](#) in Hamburg.

Dort wollten die Menschen die Gebäude vor der Spekulation schützen und haben ein ganzes Viertel besetzt. Die Stadt Hamburg hat dann die Gebäude von dem holländischen Käufer

zurückerworben und es den Menschen dort zur Verfügung gestellt.

Auch wenn es nach einer illegalen Handlung aussieht, liegt das nur daran, dass die Regierung dafür Gesetze erlassen hat. Und nur wegen dieser Gesetze ist es illegal. Wir normalen Menschen würden nicht mal auf so eine Idee kommen Gesetze zu erlassen, die uns als Menschen einschränken würden.

# Chat

Ich habe bereits begonnen eine Chat-App mit Telegramm als Vorbild zu schreiben. Nicht das ich Telegram ablösen möchte, aber ich wollte zum einen ein Beispiel haben, wie wir ein Plugin bauen können und ausserdem fände ich es gut zu wissen, das wir einen sicheren Kanal haben, mit dem wir Nachrichten austauschen können, ohne Gefahr zu laufen, das unsere Nachrichten von Großkonzernen dazu benutzt werden, Chat-Bots zu trainieren. Mehr sag ich dazu gar nicht.

# Sozial Media

Ich habe im Vorfeld schon heftig Gegenwind bekommen, wenn ich schrieb, das ich Facebook ablösen möchte, weil mir deren Zensur nicht gefällt, ich die Werbung widerlich störend finde, deren Algorithmen dazu ausgelegt sind besser Werbung platzieren zu können und sie all unsere Daten sammeln und damit ihre KI füttern.

Damit hab ich schon einige Trolle wachgerüttelt.

Aber mal ehrlich, welche Funktionen aus Facebook benötigen wir wirklich? - Eine sogenannte Wall, auf der wir unsere Fotos heften können - Einen sogenannten Feed, in dem wir unsere Gedanken und Geschichten an Freunde mitteilen können - Eine Like Funktion, um herauszufinden, ob jemand mit uns in Resonanz geht - Eine Freundesliste (haben wir schon in der App) - Und eine Gruppenfunktion

Viel mehr ist es nicht, was wir wirklich brauchen.

Den Wegfall von Zensur und Werbung brauchen wir gar nicht mal programmieren!

Folgende technische Möglichkeiten haben wir hierfür: Das Speichern der Wall und des Feeds erreichen wir mit Storj. Da jeder Nutzer einen eigenen Storage von bis zu 25 GB kostenlos zur

Verfügung hat, können diese Daten dort, natürlich verschlüsselt abgelegt werden. Nur unsere Freunde können diese Daten laden und lesen.

Wir sind dadurch natürlich nicht 100 prozentig sicher, das unsere Daten auch privat bleiben, ich denke dabei an unbewusste Freunde, die unseren Content kopieren und auf anderen Plattformen teilen, aber per se sind die Daten erstmal vor dem Zugriff dieser datenhungrigen Konzerne geschützt.

# Shop

Wenn wir nun eine eigene Währung haben und unsere Dienstleistungen anbieten wollen, dann benötigt es auch einen Marktplatz dafür. Dies überlasse ich mal eurer Kreativität. Auf jeden Fall wird es noch eine Bezahlungsfunktion in der API geben, die ihr dafür nutzen könnt.

Auch stelle ich mir vor, dass es POS (Point of Sale) Anwendungen geben wird, die als Kassensystem benutzt werden können. So eine Anwendung kann dann LMC und den EUR unterstützen und benötigt lediglich die Go Lib in der alle wichtigen Funktionen zur Verfügung gestellt werden.

Auch das Warenangebot kann auf Storc gespeichert werden.

# Desktop

Ich habe es eben bereits erwähnt, das wir auf Basis der Go Lib auch Desktop Anwendungen schreiben können.

Go läuft auf allen Plattformen und wir können Anwendungen in Swift, Kotlin oder mit Python und PySide6 schreiben und auf Go zugreifen.

Auch hier gibt es manigfaltige Anwendungsmöglichkeiten. Wie schon gesagt werden wir irgendwann ein POS benötigen.

Dann können wir ein Shopsystem um einen Admin-Teil auf dem Desktop erweitern.

Auch können wir die Go Lib in einer WASM - WebApp nutzen, wofür auch immer.



# Plugin Beispiel

Dieser Part ist nun den Entwicklern oder denjenigen, die Entwickler werden wollen, vorenthalten.

Ein Beispiel-Project findest Du in unserem Repo [shift](#) auf Github unter Plugins.

Das Plugin kann in Kotlin und wenn ich es richtig sehe auch in Java geschrieben werden. Ein Plugin basiert auf Jetpack Compose, man könnte aber sicherlich mit Hilfe von AndroidView auch old school programmieren.

## Basisc

Zu allererst müssen wir mal die `shiftapi` im `build.gradle` des Plugins nutzen.

```
1 dependencies {
2     implementation files('../..../
  android/shiftapi/build/outputs/aar/
  shiftapi-debug.aar')
```

Hierfür musst Du dann wohl den Pfad für deine Zwecke anpassen und dann wohl auch die Release-Version benutzen.

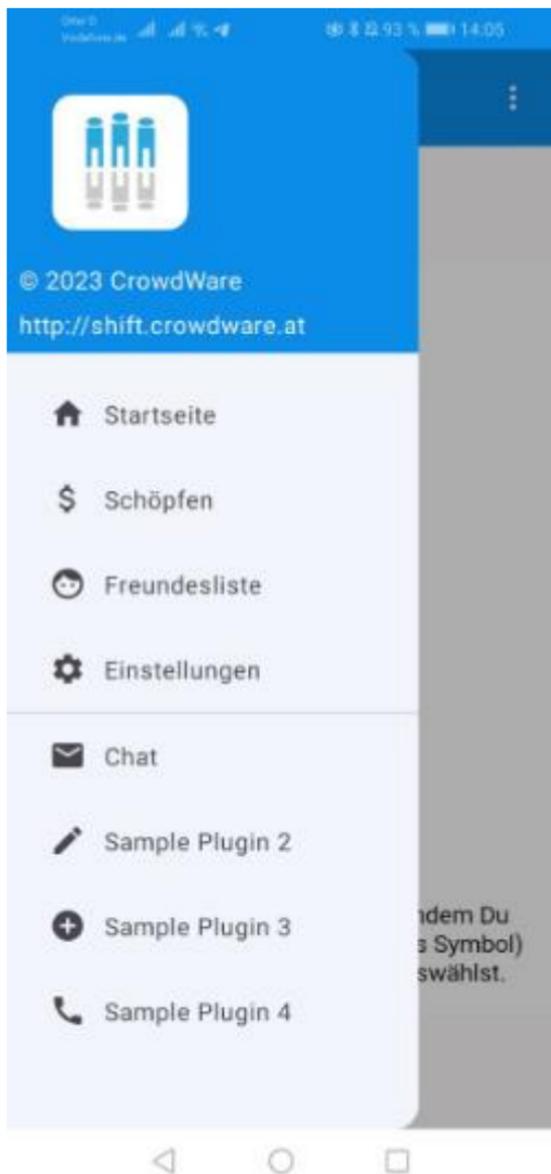
Hier nun die Implementierung des Plugins in Kotlin.

```
1 class Plugin : ShiftPlugin {
2     override fun getName(): String {
3         return "Sample Plugin"
4     }
5
6     override fun getVersion(): String {
7         return "1.0.0"
8     }
9
10    override fun menuTexts():
List<String> {
11        return listOf(
12            "Chat",
13            "Sample Plugin 2",
14            "Sample Plugin 3",
15            "Sample error")
16    }
```

In dem obigen Beispiel werden ein paar der nötigen Methoden vom ShiftPlugin überschrieben, wobei menuTexts die Menueinträge liefert, die im DrawerSheet der Main App aufgelistet werden.

Denkt bitte daran, das es später eventuell mehrere Plugins geben wird und wenn nun jedes Plugin mit 4 Einträgen wie in meinem eher schlechten Beispiel daherkommt, es für den User

eher unübersichtlich wird und er so ein Plugin sehr schnell wieder deinstallieren wird.



Wir sollten uns also am besten mit nur einem MenuItem zufrieden geben und dann auf der Seite des Plugins weiter verzweigen.

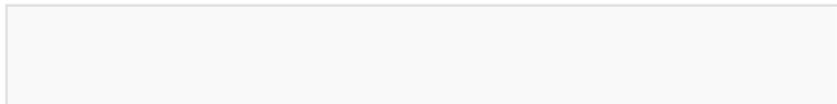
```
1     override fun icons():  
List<ImageVector> {  
2         return listOf(  
3             Icons.Default.Email,  
4             Icons.Default.Create,  
5             Icons.Default.AddCircle,  
6             Icons.Default.Call)  
7     }
```

Die Methode icons sollte zu jedem MenuItem ein Icon zurückgeben, das dann links neben dem Menu-Text angezeigt wird.

In der folgenden Methode geben wir die eigentlichen Seiten (Composables) zurück.

```
1     override fun pages(): List<  
@Composable () -> Unit> {  
2         return listOf( { Chat() }, {  
Page2() }, { Page3() }, { Page4() } )  
3     }
```

So einen Seite kann ganz simpel aussehen und einfach nur einen Text anzeigen.



```
1 @Composable
2 fun Page2() {
3     Text("Hello from plugin page 2")
4 }
```

Von hier aus ist nun deine Kreativität gefragt.

## ShiftAPI

Etwas komplizierter wird es dann jedoch, wenn Du API Funktionen ausführen möchtest. Hier ist ersteinmal der Import der API Funktion nötig und dann der Aufruf der API-Funktion.

## FriendApi

Da wir bereits Freunde und Bekannte in der Main-App einladen und hinzufügen können, schaffen wir mit der FriendApi eine Schnittstelle, um auf diese Daten zugreifen zu können.

```
1 import at.crowdware.shiftapi.FriendApi
2
3 @Composable
4 fun Chat() {
5     val friends =
6     FriendApi.getFriendList()
```

Die Funktion getFrienstList() liefert eine Liste von Friend zurück.

```
1 data class Friend(  
2     val Name: String,  
3     val Scooping: Boolean,  
4     val Uuid: String,  
5     val Country: String,  
6     val FriendsCount: Int,  
7     val HasPeerDat: Boolean)
```

**Name** des Freundes.

**Scooping** weist darauf hin, das dieser Freund in den letzten 20 Stunden das Schöpfen gestartet hat. Das ist nur dann relevant, wenn Du diesen Freund eingeladen hast Shift beizutreten. Bei Direktkontakten wird es nicht gesetzt.

**Uuid** ist die UUID des Freundes. Diese wird beim ersten Start der App vergeben.

**Country** wird nun benutzt, um den Kontinent des Peers zu speichern. (Never change a running system!)

**FriendsCount** wird auch nur gesetzt, wenn der Freund vom Nutzer eingeladen wurde. Und ist bei Direktkontakten nicht gesetzt.

**HasPeerDat** ist gesetzt, wenn der User seine Stroj-Zugangsdaten eingegeben hat. Nur wenn dies Flag gesetzt ist, kann man diesem Peer eine Nachricht senden. Dies sollte bei Direktkontakten der Fall sein, da der User hierfür die Schlüssel via QR-Code ausgetauscht hat.

Naja, zugegeben. So kompilziert ist es nun auch wieder nicht.

Aus Gründen der Einfachheit haben wir alle Funktionsaufrufe zu der Go Lib gekapselt. Da wir keine Kotlin Datentypen an Go senden können und Kotlin zum Beispiel nicht mehrere Rückgabewerte verarbeiten kann, behelfen wir uns damit, JSON Objekte also Strings auszutauschen und diese auf beiden Seiten zu entpacken.

Das klingt zwar nach einer Performance-Bremse, aber wie heißt es so schön:

"Die meiste Zeit verbraucht die UI damit, auf User-Inputs zu warten."

## Theme

Das Plugin ist ja eigentlich in den Navigation Frame eingebettet, sprich man sieht über dem Plugin Canvas die AppBar in der Primary Color.

Wenn Du nun aber Controls, wie zum Beispiel einen Button in deinem Plugin benutzt, dann solltest Du die selben Farben benutzen wie die App. Hierfür kannst Du folgende Farben nutzen:

```
1 import
  at.crowdware.shiftapi.ui.theme.Primary
2 import
  at.crowdware.shiftapi.ui.theme.Secondary
3 import
  at.crowdware.shiftapi.ui.theme.Tertiary
4 import
```

```
at.crowdware.shiftapi.ui.theme.OnPrimary
5 import
at.crowdware.shiftapi.ui.theme.OnSecondary
6 import
at.crowdware.shiftapi.ui.theme.OnTertiary
```



Tabasi debug

Hello. areetinas from ...



```
1 FloatingActionButton(
2     onClick = {},
3     containerColor = Primary,
4     contentColor = OnPrimary,
5     shape = RoundedCornerShape(16.dp),
6     modifier = Modifier
7         .padding(16.dp)
8         .align(Alignment.BottomEnd)
9     ) {
10         Icon(
11             imageVector =
12             Icons.Rounded.Add,
13             contentDescription = "Add
14             Friend",
15             tint = Color.White,
16         )
17     }
```

```
15     }  
16 }
```

## AutoSizeText

Wenn Du mal so einem Text anzeigen musst, der sich an die Größe den Canvas automatisch anpasst, dann kannst Du den `AutoSizeText` nutzen.

Wir haben den bereits auf der Schöpfen-Seite genutzt, um den Kontostand anzuzeigen.



270

```
1 AutoSizeText(  
2     if (displayBalance) {  
3  
4         NumberFormat.getNumberInstance(Locale("de",  
5         "DE")).apply {  
6             maximumFractionDigits = 3  
7             }.format(scooped.toDouble())  
8     } else {  
9  
10        NumberFormat.getNumberInstance(Locale("de",
```

```
"DE"))).apply {  
8         maximumFractionDigits = 0  
9     }.format((balance.toDouble() /  
1000.0f))  
10     },  
11     style = TextStyle(fontSize =  
70.sp, fontWeight = FontWeight.Bold),  
12 )
```

## CircularValuePicker

Den CircularValuePicker kannst Du nutzen, um einen numerischen Wert von Deinem User abzufragen, so wie wir es im HourMinutesPicker auf der RecieveGratitude-Seite machen.



```
1 CircularValuePicker(  
2     modifier = Modifier.weight(0.5f),  
3     initialValue = hours.value,  
4     maxValue = 24,  
5     unit = "h",
```

```
6     onPositionChange = { position ->
7         hours.value = position
8         total.value = (hours.value *
60 + minutes.value).toLong() +
9             LongNumber.value
10    }
11 )
```

## StorjAPI

Mit der StorjAPI können wir auf die Daten unserer Freunde zugreifen. Wir können zum Beispiel eine Nachricht auf den Storage eines Freundes ablegen, welche sich dieser Freund dann von dort aus laden kann.

Die StorjAPI benutzt eine Result-Klasse, um ein Ergebnis zurück zu geben.

```
1 sealed class ApiResponse<out T> {
2     data class Success<out T>(val
data: T) : ApiResponse<T>()
3     data class Error(val error:
String) : ApiResponse<Nothing>()
4 }
```

Ist der Funktionsaufruf erfolgreich, wird `Success` zurück gegeben, ansonsten `Error`. Wie, sehen wir gleich an einem Beispiel.

Messages haben folgendes Format.

```
1 data class Message(  
2     val Key: String,  
3     val From: String,  
4     val PeerUuid: String,  
5     val Message: String,  
6     val Time: String,  
7     val Read: Boolean)
```

**Key** ist der Pfad, unter dem die Nachricht auf Storc geschrieben wurde.

**From** ist der Absender Name.

**Time** wird wie folgt gesetzt.

**PeerUuid** ist die UUID des Empfängers.

Ist die Nachricht von heute, wird die Zeit angezeigt: **13.67**

Ist die Nachricht aus dieser Woche wird der Tag angezeigt: **Mo**

Ansonsten wird das Datum angezeigt:

**28.03.2023**

**Read** signalisiert, das die Nachricht gelesen wurde. Konkret heißt das, sie wurde geladen und vom Server gelöscht.

Das Senden einer Nachricht geschieht wie folgt:

```
1 val peerUuid = "peer_uuid"  
2 val message = "message"  
3  
4 val response =  
sendPeerMessage(peerUuid, message)
```

```

5  when (response) {
6      is ApiResponse.Success -> {
7          val ret = response.data
8          println("Message sent
successfully. Response: $ret")
9      }
10     is ApiResponse.Error -> {
11         val error = response.error
12         println("Failed to send
message. Error: $error")
13     }
14 }

```

Die PeerUuid bekommst Du aus der FriendList, siehe FriendApi.

Derzeit können wir lediglich eine Textnachricht senden.

Hier könnten wir die Api noch um binäre Dateien erweitern. Mal sehen, ob da überhaupt Bedarf besteht.

Solange die Nachricht noch nicht vom Empfänger gelesen wurde, kann sie noch gelöscht werden.

```

1  val peerUuid = "peer_uuid"
2  val messageKey = "key"
3
4  val response =
deletePeerMessage(peerUuid, messageKey)
5  if (response)
6      println("Message deleted

```

```
successfully")
7 else
8     println("Error deleting message")
```

Um die Nachrichten zu laden haben wir im ChatPlugin eine ManagerKlasse, die verdeutlichen soll, wie man die Nachrichten laden kann, gebaut.

```
1 object MessageManager {
2     var messages: MutableList<Message>
= mutableListOfOf()
3
4     fun initialize(context: Context) {
5         // TODO: This should be done in
a non blocking coroutine
6         refreshMessages()
7         val list = getMessages()
8         messages.clear()
9         for(msg in list) {
10             messages.add(Message(Key
= msg.Key, Name = msg.From, Message =
msg.Message, PeerUuid = msg.PeerUuid,
Time = msg.Time))
11         }
12     }
13
14     fun getPeerMessages():
List<Message>{
15         return messages
```

```
16     }  
17 }
```

Das ist jetzt noch nicht das BestCaseScenario, weil das Laden die UI einfrieren lässt, zeigt aber, wie man die Daten laden kann.

Mit *refreshMessages* signalisieren wir der Go Lib, das sie neue Nachrichten vom Server (Storj) holen soll. Dies ist der Teil, der einige Sekunden in Anspruch nehmen kann, je nach Netzwerk und Anzahl der Nachrichten. Würde es in einem Hintergrund-Prozess zum Beispiel alle 5 Sekunden passieren, bekommt es der User gar nicht mit. Das heißt der Bildschirm friert nicht ein und ausserdem, weiß er ja nicht, wann sein Freund auf den Senden Button gedrückt hat.

Mit *getMessages* werden dann alle Nachrichten geladen.

Zum jetzigen Zeitpunkt, haben wir da noch ein TODO, weil derzeit noch alle Nachrichten geladen werden.

Da wird es noch ein Update geben, in dem nur X Nachrichten geladen werden und man den Aufruf Pagen kann. Damit meine ich, das eventuell erstmal die Nachrichten 1-50 geladen werden und wenn der User runterscrollt dann werden die Nachrichten 51-100 geladen.

Gebt mir bitte Feedback, ob ihr diese Methoden in eurem Plugin nutzen wollt, damit wir die Funtionalität fertig implementieren können.

Solange es keiner nutzt sehe ich keinen Bedarf dies zu tun.

## Feature Request

Wir haben die API noch nicht vollständig implementiert, weil wir zu diesem Zeitpunkt noch gar nicht wissen, was wirklich gebraucht wird. Wir wissen ja noch nicht einmal ob es User geben wird, die die App überhaupt nutzen.

Mit unser Initiative wollen wir halt nur etwas bereit halten, was die Menschheit brauchen kann. Wenn ich aber an die Deutsche Politik denke und die bürgerkriegsähnlichen Zustände in Frankreich sehe, dann denke ich, es wird nicht mehr lange dauern, das die Menschen für Alternativen offen sind.

Und um ganz ehrlich zu sein denke ich, das ein cooles Plugin schon den Bedarf der App auslösen kann.

Auch ist es für Plugin-Entwickler wirklich easy einzusteigen, wenn ich daran denke, was wir alles implementieren mußten, um die App fertig zu stellen.

Der Plugin-Entwickler muss sich um Security keine Gedanken mehr machen.

Er muss auch kein Logo, Splashscreen, einen NavigationDrawer, die Localisation oder die Navigation in der App entwickeln. Das haben wir bereits alles übernommen. Auch haben wir

bereits einige Controls in der Api, die von Compose noch nicht implementiert wurden.

Wenn ihr wollt das wir ein Feature in der shiftapi implementieren, das eventuell noch andere Plugins nutzen können, dann teilt uns das bitte per Email an [artanidos@crowdware.at](mailto:artanidos@crowdware.at) mit. Während Du dieses Buch liest mache ich wahrscheinlich gerade Musik auf irgendeinem Rainbow Gathering oder helfe in einer Community. Es kann also sein, das ich Dir nicht sofort antworte, weil ich offline bin.

Ich denke da an die Implementierung der Facebook-Alternative, mit der wir die Wall und den Feed auch auf Storj ablegen könnten.

Oder vielleicht ein Dating-Plugin, wo ihr die Fotos auf Storj ablegen wollt.

Auch haben wir noch keine Schnittstelle für das Bezahlen auf einem Marktplatz.

# Über den Autor



Adam Art Ananda, wurde im November 1963 als Skorpion in Hamburg geboren. Nach Abschluss der Realschule absolvierte er eine Ausbildung zum Maschinenschlosser. Als er durch eine Wirbelsäulenerkrankung aus dem Arbeitsprozess gerissen wurde, den er sowieso nicht genoss, beschloss er, sich an der Meisterschule anzumelden. Gleichzeitig begann er einen Fernlehrgang zum Maschinenbautechniker. Art brach den Technikerkurs nach dem zweiten Semester ab, da er bereits während seines Studiums sein erstes Programm entwickelt hatte, mit dem er in kurzer Zeit viel Geld verdienen konnte.

Aus purer Neugier forschte Art weiter auf dem Gebiet der Softwareentwicklung und wurde fünf Jahre später erstmals als Berater für ein großes Chemieunternehmen eingestellt. Einige Top 500 Unternehmen waren dann seine Kunden für die nächsten Jahre, bis der Börsencrash im Jahr 2000 ihn schließlich zwang aufzugeben. Art zog dann in die Schweiz. Dort arbeitete er einige Monate für eine Fluggesellschaft und später für eine Bank. Art studierte Grafikdesign und *Human-Computer-Interaktion-Design* in der Schweiz. Letzteres hat er im dritten Semester abgebrochen, da er bereits das meiste, was dort gelehrt wird, aus seinem *Grafikdesign-Studium* wusste und (in seinem Alter) nicht mehr von einem Master-Abschluss abhängig war.

Art arbeitete einige Zeit als Tantra-Masseur, gab Sitzungen in Sexological-Bodywork, unterrichtet Menschen in der Tantramassage und gibt verschiedene andere Workshops, um Menschen zu helfen, ein besseres, aufregenderes und erfüllteres Leben zu führen.

Er engagiert sich auch für die Umsetzung der UBUNTU-Bewegung. UBUNTU („Ich bin, weil wir sind“) ist eine Idee für eine Gemeinschaft, in der es weder Geld noch Tausch noch Handel gibt. Jeder macht, was er möchte und wofür er talentiert ist. Er gibt seine Zeit für das Wohl der Gemeinschaft, in der er lebt. Im Gegenzug wird er sicherlich von der Community bekommen, was er braucht.

Art findet man in Dänemark oft beim Kitesurfen, er spielt zusammen mit ein paar Leuten im Mauerpark in Berlin Djembe, erschafft Communities in Portugal, er fuhr Motocross und Rennkart, fährt gerne Snowboard und segelt Katamarane. Außerdem fährt er lieber mit dem Fahrrad als mit dem Auto durch die Stadt und probiert ständig neue Dinge aus, die ihm gefallen könnten.

## **Künstlernername**

Um meine Freunde zu schützen, verwende ich nicht meinen bürgerlichen Namen und schon gar nicht in Großbuchstaben ;-)

### **Art**

Ich habe in dem Online Rollenspiel World of Warcraft einen Paladin mit dem Namen Arthuros gespielt und wurde von meinen Freunden schlicht Art genannt. Im Spiel habe ich mehrere Rollen ausprobiert. Den Tank, welche die Aggression der Bosse auf sich gezogen hat, um seine Gruppe zu schützen. Der Heiler, der seinen Tank geheilt hat und den Vergelter oder auch Bestrafer. Ich habe mich dazu entschlossen, gleich mal alle drei Rollen, ins Real Life zu übernehmen und kann nun jede dieser Rollen aktivieren. Ja für mich ist Realität auch nur ein Spiel.

Ein Spiel mit krass guter Grafik und tollen Effekten wie der Geruch und das haptische ;-)

## **Ananda**

Kennst Du Siddhartha von Hermann Hesse? Genauso sehe ich es auch. Ich folge keinem Lehrer oder Guro, denn ich bin mein eigener Guru. Und deshalb habe ich mir selbst einen Sannyasin Namen gegeben. In meinem Fall Ananda, die *Abwesenheit von Unglück* das aus dem Sanskrit stammt.

## **Adam**

In Portugal habe ich mit einem lieben Menschen aus Holland ein Ritual gemacht, ich wollte sie gerne als meine Partnerin gewinnen, und habe ihr einen Apfel spendiert. Und an das Paradies erinnert. Den Sündenfall, als Adam und Eva den Apfel vom Baum der Erkenntnis aßen. Der Apfel begleitet mich schon mein ganzes Leben. Wir hatten zwei Apfelbäume im Garten. Diese haben mich erst vor ein paar Jahren daran erinnert, das wir eigentlich kein Geld für Lebensmittel ausgeben müßten, denn Mutter Erde liefert uns doch genug zu Essen, ohne, das wir dafür etwas tun müssen.

Auch waren auf den Schallplatten der Beatles ein Hinweis auf den Apfel. Die Platten waren von Apple-Records bedruckt worden. Man sah auf Seite 1 einen Grünen Apfel und auf Seite zwei einen aufgeschnittenen Apfel. Dieses Buch schreibe ich auf einem Mac von Apple Machintosh. Das iPhone motivierte mich Apps zu programmieren, weil ich es nützlich fand. So kannte ich mich mit dieser Materie aus und konnte die App für Android schreiben.

Da ich bei mir angefangen habe Traumata aufzulösen und anderen Menschen auch dabei helfe und auch Glaubenssätze löschen, bzw. umprogrammieren kann, sehe ich mich in der Lage, selbst wenn ich damit wohl verlacht werden würde, auch die Traumata unser Vorfahren bis hin zu Adam und Eva aufzulösen. Das geht zum einen mit Vergebung und zum anderen auch sich selbst zu vergeben, in der Opferrolle gewesen zu sein. Ein Versuch ist es allemale Wert. Hab ja noch genug Zeit. Auch wenn wir nur daran denken, jemanden zu erwürgen, schaffen wir auf diese Weise Karma. Und dieses Karma können wir wieder abbauen. Mit einer Technik, die ich im Yoga gelernt habe. Die sich Konsekrieren nennt. Also eine Widmung an das Universum.

# Bücher

**Python GUI** - Erstelle Cross-  
Plattform Desktop Applikationen mit Python, Qt und PySide6 Olaf Japp

<https://www.amazon.de/dp/B0BXYPZ6VY>

**Manifestiere ein besseres Leben** - Adam Art  
Ich mach mir die Welt, widde widde Ananda  
wie sie mir gefällt

<https://www.amazon.de/dp/B08CB8Q54L>

# Video

**Das Ei** Andy Weir

<https://youtu.be/VmHAuMGJwe4>

**Music** Rainbow Family

<https://youtu.be/24qveCTn4KE>

# Quellcode

## URL

Front End	<a href="https://github.com/CrowdWare/shift">https://github.com/CrowdWare/shift</a>
Business Logik	<a href="https://github.com/CrowdWare/shift-go">https://github.com/CrowdWare/shift-go</a>
Web Service	<a href="https://github.com/CrowdWare/shift-webservice">https://github.com/CrowdWare/shift-webservice</a>

# Software

Dieses Buch wurde mit Hilfe von EbookCreator  
<https://ebc.crowdware.at/>  
geschrieben.

Der EbookCreator ist eine InHouse-Anwendung,  
die es erlaubt, den ganzen Text mittels  
Markdown einzugeben. Der EbookCreator  
erstellt auf Knopfdruck eine EPUB Datei, welche  
zum Beispiel auf Amazon KDP  
<https://kdp.amazon.com>  
verwendet werden kann um ein Ebook dort zu  
veröffentlichen.

Des Weiteren kann das Buch auch als HTML  
gespeichert werden, mit weasyprint in PDF  
umgewandelt um bei Tolino ein Paperback oder  
Hardcover zu veröffentlichen.

# Vergebung

Ich vergebe allen Menschen die mir durch ihre Taten gezeigt haben, das Gewalt keine Lösung ist.

Ich vergebe allen Opfern, die sich in die Opferrolle gebracht haben.

Ich vergebe mir, das ich auch gewaltbereit war.

Ich vergebe mir, das ich in die Opferrolle gegangen bin.

Es tut mir leid

Bitte verzeih mir Ich liebe Dich Danke

# Träume

Ja, ich bin ein Träumer!  
Und das ist auch gut so! Einer muss es ja machen ;-)

Träume können wir erst dann  
verwirklichen, wenn wir uns  
entschließen, daraus zu erwachen.  
- Martin (Gott stehe ihm bei)

Ich weiß jetzt, das wenn ich damit fertig bin,  
dieses Buch zu schreiben, ich aufwachen werde  
und mich wohlmöglich nicht mehr daran erinnern  
kann, weil mein Wecker geklingelt hat.

Ich werde dann eventuell gleich mal den Wecker,  
den Fernseher, das Rundfunkgerät die Klamotten  
mit Firmenlogos, mein Handy und meinen  
Computer aus dem Fenster werfen und kann  
dann wieder frei durchatmen und werde ab  
diesem Tage wieder schöne Dinge erträumen.

Am Ende ist alles gut und wenn am  
Ende nicht alles gut ist, dann ist es  
noch nicht das Ende.  
- Unbekannt

# Loslassen

Ich zeige euch nun mal eine Übung im Loslassen.

Theopraktisch könnte ich die App ja für ein paar Euros verkaufen. Sagen wir mal 1,- EUR. Das könnte mich über Nacht zum Millionär machen, in dem ich die App ne Million mal verkaufe.

Mach ich aber nicht, denn was soll ich mit diesen wertlosen Euros die aus dünner Luft geschöpft werden?

Ich wurde heute von einem anscheinend reichen Mann gefragt, ob er mir finanziell helfen kann mit der App.

Ich sagte, "nee danke ich habe genug, ich bekomme Bürgergeld."

Du kannst Dir nicht vorstellen, was ich in seinen erstaunten Augen gesehen habe.

Das war so eine Art "Wat fürn Spinner", bis zu, "Oder, oh, der meint es ernst, der will wirklich das Geld abschaffen!"

Genau das ist mein Ziel.

Ich bin gerade in so einen Prozess gekommen, weil mir vor ein paar Tagen Gewalt vorgespiegelt wurde und ich außerdem von einem lieben Freund gezeigt bekommen habe, was man mit Geld alles kaufen kann. Wir wollten uns in Berlin Schöneberg bei der 12 Apostel-Kirche treffen und

dazu ging ich zufuss über Gleisdreieck und durch die Kurfürstenstrasse. Dort wurde ich ganz nett von den Damen mit der Frage "Haste Lust" begrüßt. Ich antwortete, "na klar" und ging weiter. Und bei dem Thema Gewalt meine ich natürlich Waffen, Drogen, Politiker, Zeitungen, Fernsehen und all die anderen Dinge, die man kaufen kann.

Und während meines Prozesses habe ich die App ganz einfach loslassen können, weil ich jetzt weiß, das wir das Geld auch ohne die App loswerden können.

Wir können uns auch ohne selbstgeschöpftes Geld gegenseitig Dankbarkeit ausdrücken. Das geht mit einer einfachen Verbeugung, einem Handschlag, einer Umarmung oder mit den Worten "Dankeschön".

Die App wird veröffentlicht, aber das Schöpfen bleibt deaktiviert und dient nur noch als Mahnmal.

Der Initialbetrag kann weiterhin genutzt werden, um die Funktionen auszuprobieren oder was auch immer.

Zumindest diejenigen, die dieses Buch nicht lesen werden, werden auch nicht schöpfen können. Da Du es ausgehalten hast, bis hierhin zu lesen, verrate ich Dir ein Geheimnis.

I love UBUNTU

Nutze dieses in der App und auch Du wirst zum Schöpfer ;-)

Ich habe vor ein paar Jahren in Dänemark gewohnt und konnte dort an einem sagen wir mal tantrischen Ritual teilnehmen, das in der Sylvesternacht stattgefunden hat.

Ja wir saßen in der Maithuna-Stellung (Jub Yum) aber mit Klamotten an, denn die Energie fließt ja auch durch den dünnen Stoff.

Um uns in die nötige Energie zu bringen, haben wir uns vorher mit Hilfe der QLB, eine sehr starke Atmungs-Meditation, eingeschwingt.

Und dann unseren Wunsch ins Universum gesendet.

Mit meinem Wunsch habe ich den Love-Coin manifestiert.

Nein, das ist kein Geld und auch keine Münze, sondern nur eine Idee, wie wir uns Dankbarkeit schenken können.

Mit Liebe 

Heute benötige ich kein Einkommen mehr, da ich bereits ein Auskommen habe.

Und da Du jetzt immer noch liest, hast auch Du den Schlüssel zum Glück in der Hand.

Ich lass das Ende mal offen, damit Du Deine  
Träume selber träumen kannst...

Mein Traumtagebuch:

Ich hoffe, das wir uns eines Tages in einer dieser neu entstehenden Dörfer wiedersehen und über diesen fantastischen Traum reden können.

Art