Krishna Kumbhar

Project handover plan

Version <1.0>

Release history

| **Date** | **Version** | **Comment** | **Author****(Developer)** | **Approved By****(TL)** |
| --- | --- | --- | --- | --- |
| 12/05/2021 | 1.0 | Gamelancer | Krishna KumbharAakash Narkar, Mukul Murya | Nivrutti Patil |
| 01/04/2018 | 1.0 | Havanote | Krishna KumbharAbhishek KohliAaskah Narkar | Nivrutti Patil |
| 10/07/2020 | 1.0 | DiPulse | Krishna KumbharNivrutti PatilAyon GuptaHarshali TamhankarBhushan Bafna | Nivrutti Patil |
| 31/05/2021 | 1.0 | MedPenny | Krishna KumbharChhaya PatelTanmay Khopkar | Nivrutti Patil |
| 01/04/2021 | 1.0 | iLA Bank | Krishna KumbharPushpa ShirkoleArvind ValakiVivek Athilkar | Nivrutti Patil |
| 01/02/2020 | 1.0 | Covid Proximity | Krishna Kumbhar | Nivrutti Patil |

Index

1. Project background 4

1.1 Client 4

1.2 Purpose 4

1.3 Scope 4

1.4 Work role 4

2. Project description 4

2.1 Core work flow 4

2.2 Architecture 4

2.3 Work item 4

2.4 Project Team 4

3. Handover plan 4

3.1 Handover timeline 4

3.2 Handover method 4

4. Notice 4

4.1 Cooperation model 4

4.2 Frequently Asked Questions 5

5. Account information 6

5.1 Site 6

5.2 Source code 6

5.3 Resources 6

5.4 App iTunes/Play Store URL 6

6. Work Status 6

6.1 Current Status 6

6.2 Status project on last day of contract 6

Project handover plan

# Gamelancer

# Project background

*Gamelancer is the biggest platform for gaming users. This application is similar to the gaming discord platform. In this application, we can create an account and log in. After login, there are lots of gaming rooms where users can enter and play with other users. User can create their own rooms and invite other players to play together. Rooms can be private or public. Users can add, blocked other users in this application. Project is Completely build on swiftUI,Agora Sdk is used for live chatting and vedio sharing in this project.*

## 1.1 Client

*Eyenov*

## 1.2 Purpose

*Platform for gamers to interact with other team members*

## 1.3 Scope

*Social platform for gamers*

## 1.4 Work role

*App features development with SwiftUI language*

# Project description

## 2.1 Core work flow

*Application built using SwiftUI*

## 2.2 Architecture

*MVVM*

## 3. Work item

*Application development is closed from our end and code is shared with client*

## 4. Project Team

*Krishna Kumbhar
Aakash Narkar
Mukul Maurya*

# Handover plan

## 3.1 Handover timeline

*1 Week*

## 3.2 Handover method

*KT is done with Co-Developers*

# Notice

## 4.1 Cooperation model

*None*

## 4.2 Frequently Asked Questions

*None*

# 5. Account information

## 5.1 Site

*[The admin portal/client website URL, account and Attach description if necessary]*

## 5.2 Source code

*Source code was hosted on client version control tools*

## Resources

*All required resources were hosted on client cloud storage.*

## App iTunes/Play Store URL

*App is not on store yet*

# Work Status

## 6.1 Current Status

*Terminated*

## 6.2 Status project on last day of contract

*Contract is closed*

# Havanote

# Project background

*Havanote is one of biggest in the air platform for users which allows to create notes in a location and tag it to self or bunch of people so that whenever respective user visits note location get immediate notifications and and allows user to discover contents dropped by others users in nearby area.*

*Here user and have direct message to other users for one to one or many communication. User can track complete travel journey through location log wit custom settings, also able to add favorite locations so on every visit on users favorite location if there are any new notes then user will get notified which is handled with live location tracking and region monitoring.*

*In havanote easy to drop features also available with help of share extension and 3d touch to create direct connect in app.*

*On developement front we are using latest version of swift with MVVM design pattern, RX-Swift, Google maps, Firebase, Bind, etc.*

## 1.1 Client

*Shekar Raman*

## 1.2 Purpose

*Location Based Social Media*

## 1.3 Scope

*Location, Social Media,*

## 1.4 Work role

*App features development with Swift language*

# Project description

## 2.1 Core work flow

*Application built using Swift language and MVVM architecture*

## 2.2 Architecture

*MVVM*

## Work item

*Application development is closed from our end and code is shared with client*

## Project Team

*Krishna Kumbhar
Abhishek Kohli
Aakash Narkar*

# Handover plan

## 3.1 Handover timeline

*1 Week*

## 3.2 Handover method

*KT is done with Co-Developers*

# Notice

## 4.1 Cooperation model

*None*

## 4.2 Frequently Asked Questions

*None*

# 5. Account information

## 5.1 Site

*https://www.havanote.com*

## 5.2 Source code

*Source code was hosted on client version control tools*

## Resources

*All required resources were hosted on client cloud storage.*

## App iTunes/Play Store URL

*https://apps.apple.com/in/app/havanote/id1448212759*

# Work Status

## 6.1 Current Status

*Terminated*

## 6.2 Status project on last day of contract

*Contract is closed*

# DiPulse

# Project background

*diPulse is a fitness app based on iOS core bluetooth framework which is used for body building, toning, relaxation, cramp prevision etc. You just need to connect BLEs to the app using bluetooth, select the workout which you want to perform and start the workout. BLEs generate the stress at muscular level which increases over all strength of the body. App also has the separate heart rate module which measures heart rate and plots the graph of the same throughout the workout session. You can update firmware of the BLE using the app.*

*Features*

*In-App Purchase of product Subscription*

*Customize heart dashboard as per user preferences (Analog, Digital graph representation.)*

*Multiple language Support like (English, Chinese, French, German, Russian)*

*Sync data with Health Kit*

*iWatch support*

*Type Native iOS App*

*Programming Language Used SWIFT*

*AppStore URL https://apps.apple.com/in/app/dipulse/id1472064553*

*iOS Version Supported 10.0, 11.0, 12.0, 13.0*

*Approximate Number Of Screens 100*

*Supported Devices Apple Watch,, iPhone,*

*Orientation Support Portrait*

*Database Used CoreData*

*Web Services Used REST*

*Mode Online/Offline*

*Localization Supported YES*

## 1.1 Client

*DiPulse*

## 1.2 Purpose

*Fitness and Healthcare with hardware support*

## 1.3 Scope

*Fitness, Bluetooth*

## 1.4 Work role

*App features development with RX-Swift language*

# Project description

## 2.1 Core work flow

*Application built using RX-Swift language and Clean architecture*

## 2.2 Architecture

*Clean Architecture*

## Work item

*Application development is closed from our end and code is shared with client*

## Project Team

*Krishna Kumbhar
Nivrutti Patil
Ayon gupta
Bhushan Bafna
Harshali Tamhankar*

# Handover plan

## 3.1 Handover timeline

*1 Week*

## 3.2 Handover method

*KT is done with Co-Developers*

# Notice

## 4.1 Cooperation model

*None*

## 4.2 Frequently Asked Questions

*None*

# 5. Account information

## 5.1 Site

*https://dipulse.com*

## 5.2 Source code

*Source code was hosted on client version control tools*

## Resources

*All required resources were hosted on client cloud storage.*

## App iTunes/Play Store URL

*https://apps.apple.com/in/app/dipulse/id1472064553*

# Work Status

## 6.1 Current Status

*Terminated*

## 6.2 Status project on last day of contract

*Contract is closed*

# MedPenny

# Project background

*A mobile application which will allow individual customers to save the spare changes that are left after making a payment from their bank accounts and make use of that amount to Co-pay to health service providers (HSP). For health service providers, it will be a web panel from which they can receive money from individual customers for their co-pay.
Core Graphics, UIKit, UserNotification, Alamofire, Firebase Analytics, Firebase PushNotification, IQKeyboardManager*

## 1.1 Client

*Medpenny*

## 1.2 Purpose

*Collect small penny from transactions leftover amount and use that for medical expences*

## 1.3 Scope

*Healthcare and Transactions*

## 1.4 Work role

*App features development with Swift language*

# Project description

## 2.1 Core work flow

*Application built using Swift language and MVVM architecture*

## 2.2 Architecture

*MVVM*

## Work item

*Application development is closed from our end and code is shared with client*

## Project Team

*Krishna Kumbhar
Chhaya Patel
Tanmay Khopkar*

# Handover plan

## 3.1 Handover timeline

*1 Week*

## 3.2 Handover method

*KT is done with Co-Developers*

# Notice

## 4.1 Cooperation model

*None*

## 4.2 Frequently Asked Questions

*None*

# 5. Account information

## 5.1 Site

*-*

## 5.2 Source code

*http://mobilegit.neosofttech.in/IOS/MedPenny\_iOS*

## Resources

*All required resources were hosted on client cloud storage.*

## App iTunes/Play Store URL

*App is still under development*

# Work Status

## 6.1 Current Status

*OnGoing*

## 6.2 Status project on last day of contract

*Contract is not closed yet*

# iLA Bank

# Project background

*ila is a digital, mobile-only bank based in Bahrain, that’s built around your needs and aspirations. We understand how you use technology to help you make the most out of your money.*

*‘ila’ is the Arabic word for ‘to’ and we are here to truly reflect your needs and lifestyle, evolving with you on your journey.*

*We understand how you use technology to help you make the most out of your money, ‘ila’ is the Arabic word for ‘to’ and we are here to truly reflect your needs and lifestyle, evolving with you on your journey.*

*- Join ila in minutes*

*All you need is 1 ID and a selfie, there is no need to visit a branch or sign any documents.*

*- Instantly issued Virtual card*

*You will get your virtual card in your app as soon as your account is open, and you can start using it for all your online transactions.*

*- A cool green card*

*Enjoy using your green ila debit card anywhere in the world. Use contactless payments to speed up your transactions, making your purchases extra secure as the card never leaves your hand.*

*- Bahraini Dinar account with tiered interest*

*The money in your account earns you interest, interest that’s tiered. The more you save, the more you earn.*

*- Open and link foreign Currency Accounts to your card*

*Open Foreign Currency accounts in two easy steps from your app and link multiple currencies to your cards. Spend in local currencies and have the freedom to explore the world.*

*- Easy ways to fund your account*

*Simply use your existing bank debit card, BenefitPay, bank transfer, or fund your account by depositing cash or cheque at an ila ATM.*

*- Rewarding Credit Cards*

*Apply for a Credit Card directly from the app and enjoy 100% control. Get 1% cashback on all spends and a chance to win Gold every month for every BHD 1 you spend!*

*- Save and win BIG with Al Kanz*

*Deposit money in your Al Kanz Account to get a chance to win BIG. Every BHD 50 is one chance to win cash prizes up to USD 1 Million!*

*- Start saving with Hassala*

*Use the Hassala feature to start saving for the things that matter most. Customize your Hassala with a name and icon to keep track of your progress. Set goals with a target amount and date, automate payments from any currency to your Hassala and celebrate once you achieve your goal.*

*- Start saving groups with Jamiyah*

*With Jamiyah, you and a group of your trusted friends or family can work together to create a community saving. Each participant will pay a monthly contribution, and when it’s their turn, they will get the entire savings in one month!*

## 1.1 Client

*Mohammed Almraj*

## 1.2 Purpose

*Banking services provider*

## 1.3 Scope

*Banking Domain*

## 1.4 Work role

*App features development with Swift language*

# Project description

## 2.1 Core work flow

*Application built using Swift language and MVVM architecture*

## 2.2 Architecture

*MVVM*

## 3. Work item

*Application development is going on and source code is hosted on client VC*

## Project Team

*Krishna Kumbhar
Pushpa Shirkole
Arvind Walaki
Vivek Athilkar*

# Handover plan

## 3.1 Handover timeline

*1 Week*

## 3.2 Handover method

*KT is done with Co-Developers*

# Notice

## 4.1 Cooperation model

*None*

## 4.2 Frequently Asked Questions

*None*

# 5. Account information

## 5.1 Site

*-*

## 5.2 Source code

*Hosted on client version control*

## Resources

*All required resources were hosted on client confluence server.*

## App iTunes/Play Store URL

*https://apps.apple.com/bh/app/ila-bank/id1480238512*

# Work Status

## 6.1 Current Status

*OnGoing*

## 6.2 Status project on last day of contract

*Contract is not closed yet*

# Covid Proximity

# Project background

*The users will have the ability to check themselves into that relevant venue in the app, with the objective of maintaining social distance throughout your time of being at that venue, and this is achieved by alerting users if anyone comes within a 2m proximity of them.*

*That will then set off an alarm on their phone to warn them that they have come within that 2m distance and that should prompt them to step away and continue adhering to the social distancing rules in place.*

*Core features of app:*

*- Show proximity alerts to user when new user come into 2m proximity.*

*- Add your closed one's into bubble to exlude alerts for bubble users*

*- Check-in and Check-out from event*

*- Proximity alerts in background*

*- I am alive push notification for user to make app active*

*- Location Based venue search*

## 1.1 Client

*Pocket App*

## 1.2 Purpose

*Nearby User Detection using bluetooth technology*

## 1.3 Scope

*Social Distancing*

## 1.4 Work role

*App features development with Swift language*

# Project description

## 2.1 Core work flow

*Application built using Swift language and MVVM architecture*

## 2.2 Architecture

*MVVM*

## Work item

*Application development is done and source code is hosted on client VC*

## Project Team

*Krishna Kumbhar*

# Handover plan

## 3.1 Handover timeline

*-*

## 3.2 Handover method

*Application development is closed and necessary handover is shared with client end developers*

# Notice

## 4.1 Cooperation model

*None*

## 4.2 Frequently Asked Questions

*None*

# 5. Account information

## 5.1 Site

*-*

## 5.2 Source code

*Hosted on client version control*

## Resources

*All required resources were hosted on client confluence server.*

## App iTunes/Play Store URL

*-*

# Work Status

## 6.1 Current Status

*Closed*

## 6.2 Status project on last day of contract

*Contract is closed*