

MagneFlex Browser

Magensa Services Developer Tool Installation and Operation Manual

August 2021

Manual Part Number:

D998200177-50

REGISTERED TO ISO 9001:2015

Magensa Services Developer Tool - MagneFlex Browser

Copyright © 2006 - 2021 MagTek, Inc.

Printed in the United States of America

Information in this publication is subject to change without notice and may contain technical inaccuracies or graphical discrepancies. Changes or improvements made to this product will be updated in the next publication release. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of MagTek, Inc.

MagTek® is a registered trademark of MagTek, Inc.

MagnePrint® is a registered trademark of MagTek, Inc.

Magensa™ is a trademark of MagTek, Inc.

MagneSafe® is a registered trademark of MagTek, Inc.

DynaPro™ and DynaPro Mini™ are trademarks of MagTek, Inc.

IPAD® is a trademark of MagTek, Inc.

iPhone®, iPod®, and Mac® are registered trademarks of Apple Inc., registered in the U.S. and other countries. App StoreSM is a service mark of Apple Inc., registered in the U.S. and other countries. iOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used by Apple Inc. under license.

iPad™ is a trademark of Apple, Inc.

The *Wi-Fi*® is a *registered trademark* of *Wi-Fi Alliance*.

Bluetooth® is a registered trademark of Bluetooth SIG.

Microsoft® and Windows® are registered trademarks of Microsoft Corporation.

All other system names and product names are the property of their respective owners.

CONFIDENTIAL

This document may not be reproduced or distributed. This document is for informational purposes only. Changes to this document may occur without notice.

Magensa Services Developer Tool - MagneFlex Browser

Table 0.1 - Revisions

Rev Number	Date	Notes
10	May 2, 2017	Initial Release
20	July 8, 2017	Added screen showing favorites and pop-up management
30	Jan 22, 2018	Added Android App
40	Oct 9, 2020	Updated Appendix A Updated with MF 2.0 (Windows) screens and some other organization for consistency
50	August 2021	Added support for iDynamo 6 via USB Reader. Updated URL for MagneFlex Sample Code Demo

Table of Contents

Table of Contents 4

1 Introduction 5

2 System Architecture 6

3 Browser Functions 7

 3.1 Windows 7

 3.2 Apple iOS 12

 3.3 Android 17

4 HTML Application Integration 23

 4.1 Operation Definitions and Inputs 25

 4.2 Operation Outputs 31

5 Test page 33

6 Appendix A – Supported Readers 34

7 Appendix B – Common MagTek Magnetic Stripe Reader TLV Tags 35

1 Introduction

MagneFlex Browser is a full-function HTML browser application for Windows, iOS, and Android. It is integrated with driver applications that allow the control of a MagTek payment card Reader from an HTML over HTTP(S) context. Integrators can build an HTML Application, such as a web-based POS Application, that can interact with MagTek Readers through the MagneFlex Browser to collect data from magnetic stripe, contact chip, or contactless (NFC) cards for the purpose of processing a payment. MagneFlex Browser can be distributed by the integrator to its end users, along with MagTek Readers, to deploy a full-function POS system.

2 System Architecture

MagneFlex Browser employs an architecture that contemplates the following elements of a payment system:

- **Reader** – A supported MagTek payment card Reader. See Appendix A or a MagTek sales representative for the current list.
- **User Device** – The computer system that is connected to the Reader. The User Device is also where the MagneFlex Browser application is installed. Supported operating systems include Windows (version 7 or later), Apple iOS, and Android. Not all Readers can be supported by all operating systems. Please see a MagTek sales representative for further details.
- **HTML Application** – The application, provided by the integrator of MagneFlex Browser that will communicate with the Browser to control and obtain data from the MagTek Reader. This is often a POS Application for processing payment transactions but is not limited to this use. MagTek Readers can be used for many other purposes, including multi-factor authentication, facilities access, closed-loop loyalty and payment applications, etc. Magensa, MagTek’s payment processing subsidiary, has a wide variety of services that can be used to process data from the Reader downstream to payment processors or other endpoints, while maintaining the point-to-point encryption (MagneSafe) provided by MagTek Readers. Please see a MagTek sales representative or www.magensa.net for more details.
- **MagneFlex Browser** – A full-function HTML browser application, for Windows, Apple iOS, and Android, that is integrated with driver applications that allow the control of a set of MagTek payment card Readers from an HTML over HTTP(S) context.

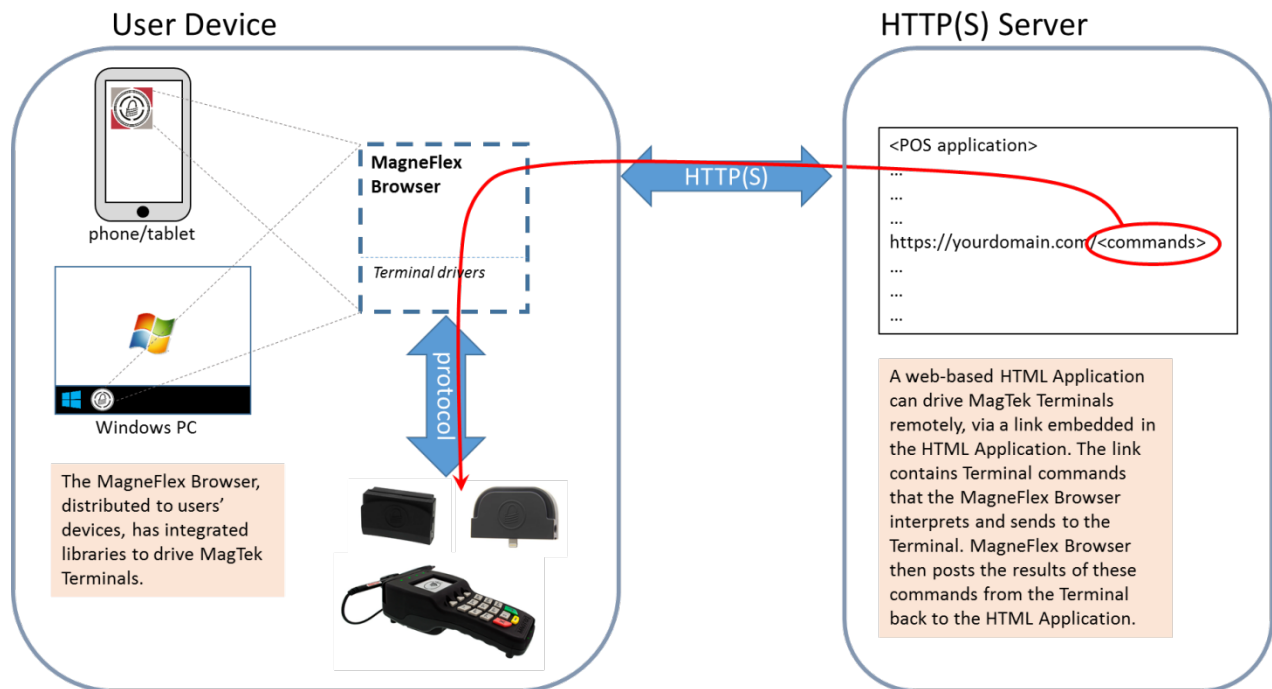


Figure A: Diagram of MagneFlex Browser

3 Browser Functions

The MagneFlex Browser includes all the normal functions found in popular HTML browsers. Additional features have been added to assist in managing Readers connected to the User's Device. The following section outlines the major features of MagneFlex Browser by operating system.

3.1 Windows

Navigation controls are common with standard browsers. Settings are also common, except for the type of Reader connected. If you are connecting a MagTek SCRA or eDynamo, choose MSR (USB). If you are connecting to a DynaPro family Reader, select PinPad (USB) or PinPad (Ethernet – an IP address will be required). The MagneFlex Browser may be customized by the HTML application developer, including buttons and trim colors. Please contact a MagTek sales representative for further details.

The main menu for the Windows version of the MagneFlex Browser can be accessed by clicking on the vertical ellipsis or kebab menu (:) at the top right of the Browser window.

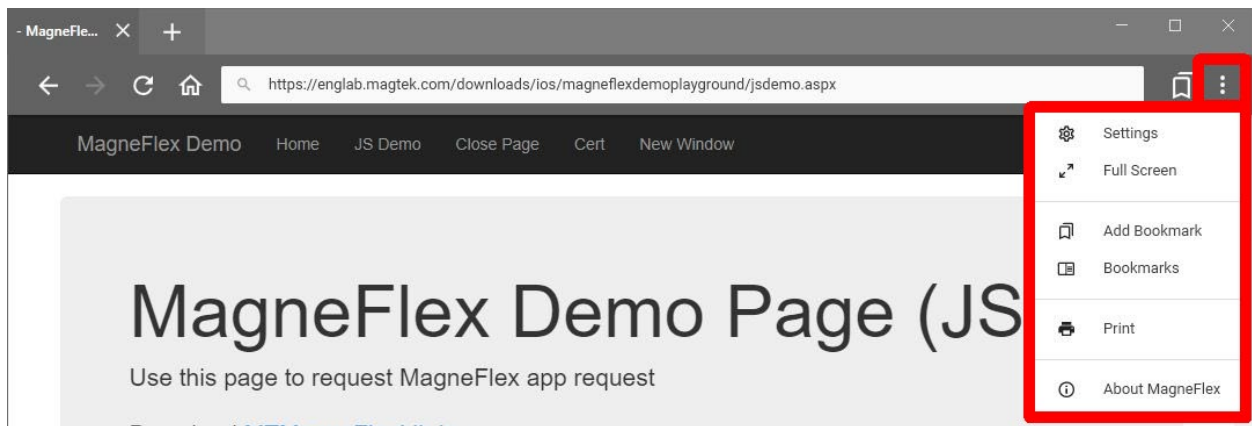


Figure B: Main menu, MagneFlex Browser for Windows

The following sections describe the options available from this menu.

3.1.1 Settings

Click on **:** → **Settings** to open the window from which the MagneFlex Browser settings can be configured.

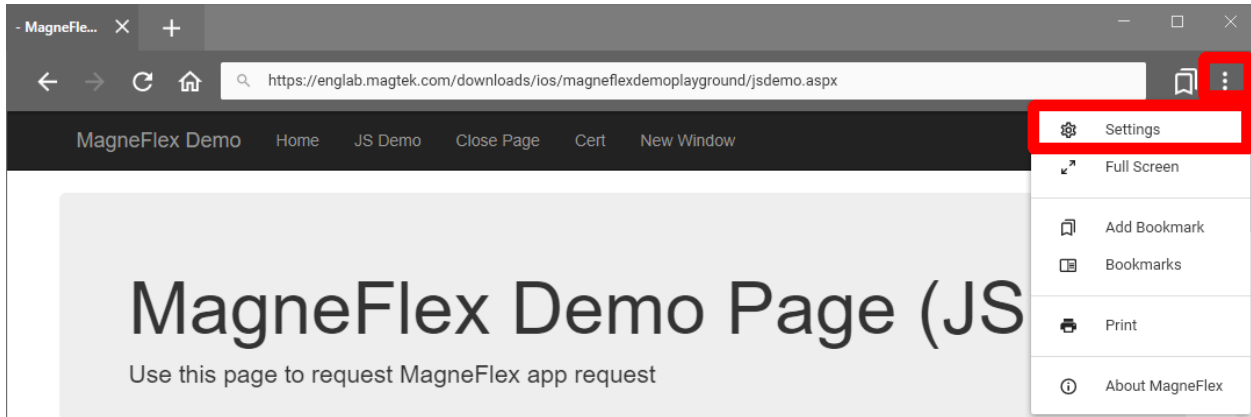


Figure C: Settings menu, MagneFlex Browser for Windows

To access the menu settings available from this window, click on the hamburger menu (≡) icon in the upper left corner of the window.

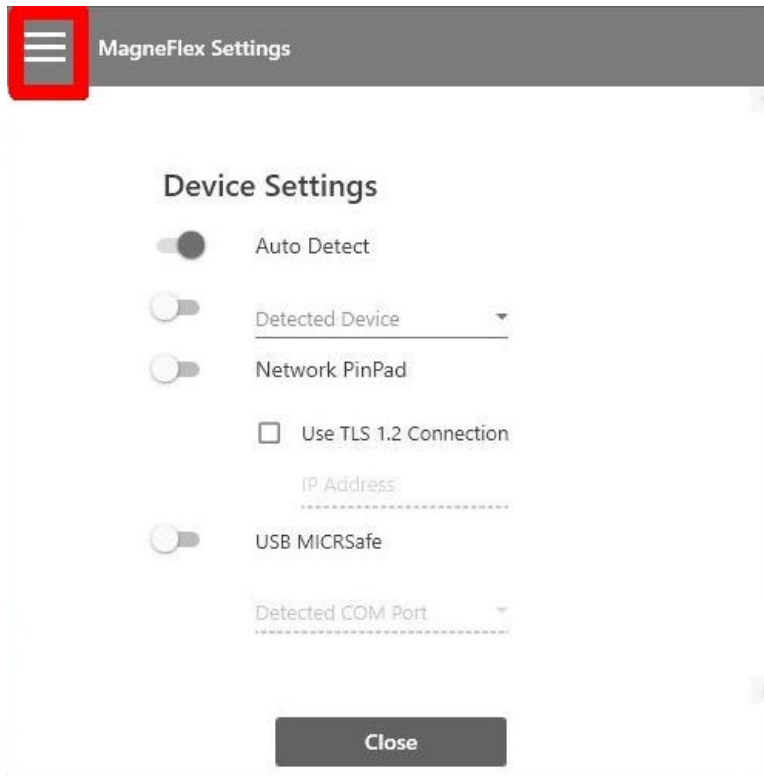


Figure D: Settings submenu, MagneFlex Browser for Windows

3.1.1.1 Device Settings

Click on ≡ → **Device Settings** to modify the settings to be used for the MagTek Reader that will be accessed by the MagneFlex Browser.

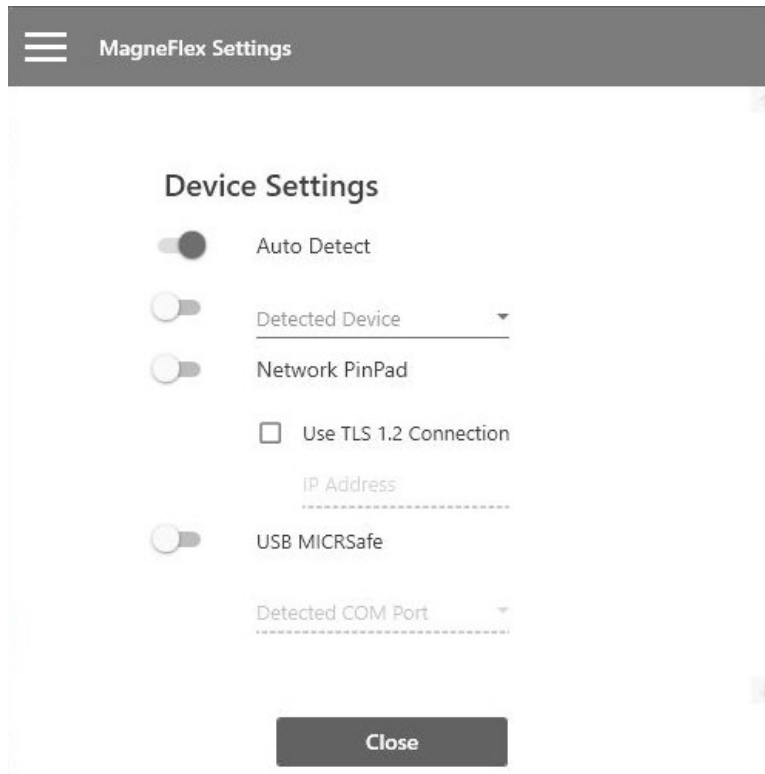


Figure E: Device Settings, MagneFlex Browser for Windows

3.1.1.2 Launch Settings

Click on ≡ → **Home** to access the settings that determine what the MagneFlex Browser will do when it first runs.

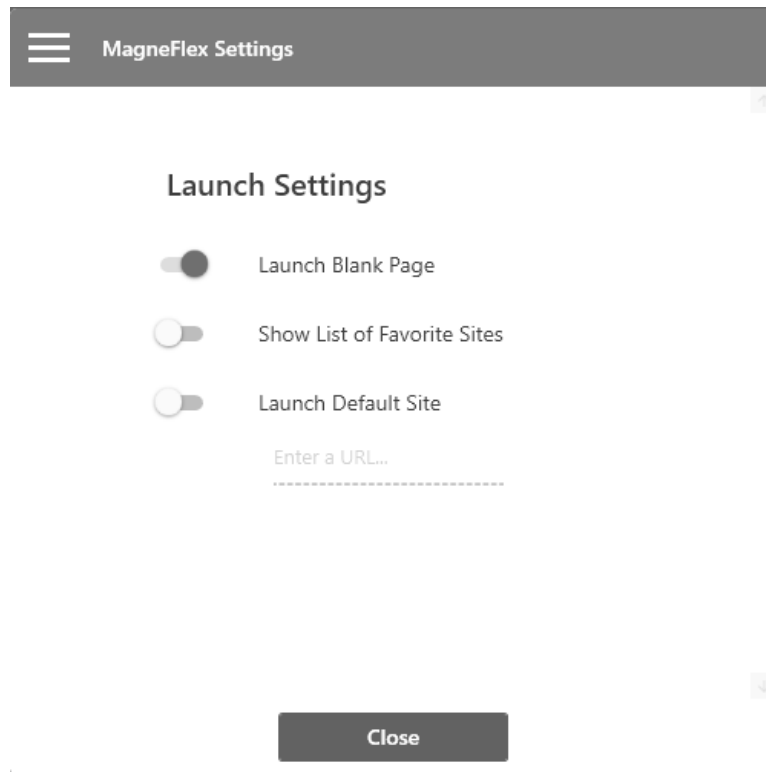


Figure F: Launch Settings, MagneFlex Browser for Windows

3.1.1.3 Privacy Settings

Click on ≡ → **Privacy and Security** to change Browser cache settings, set whether the cache is cleared on program exit, control whether pop-up windows are permitted, and clear the cache.

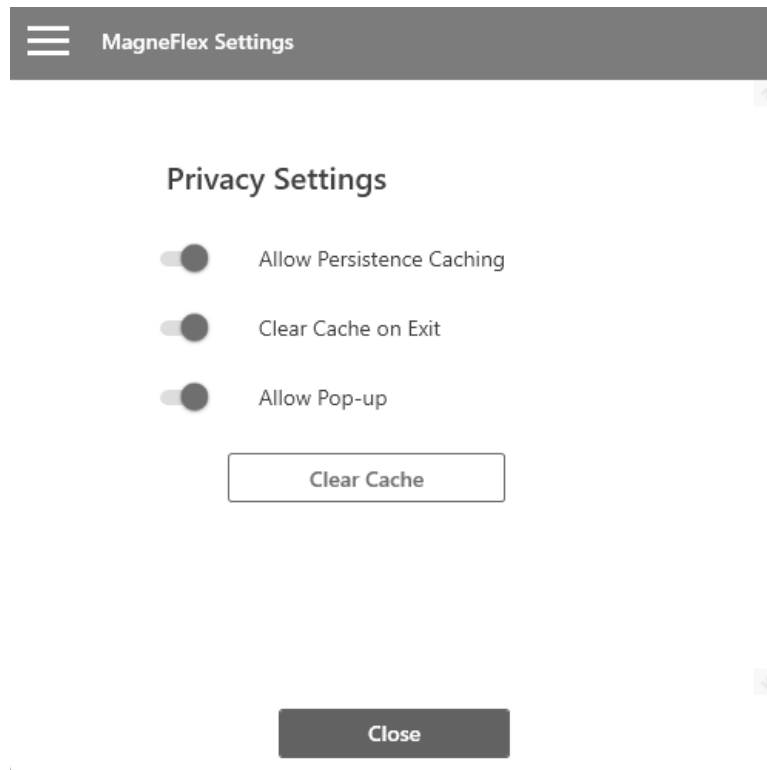


Figure G: Privacy Settings, MagneFlex Browser for Windows

3.1.2 Full Screen

Click on : → **Full Screen** to put the MagneFlex Browser into full screen mode. Press the [F11] key to switch back to windowed mode.

3.1.3 Add Bookmark

Click on : → **Add Bookmark** to add a bookmark.

3.1.4 Bookmarks

Click on : → **Bookmarks** to see the list of the bookmarks you have saved in the Browser.

3.2 Apple iOS

MagneFlex Browser is available in the Apple App Store. Search “MagneFlex”. This application is also available for customization and can be deployed to the Apple App Store under the HTML Application developer’s own Apple product plan. Please see a MagTek sales representative for further details.

The following sections describe the options available from the main menu.

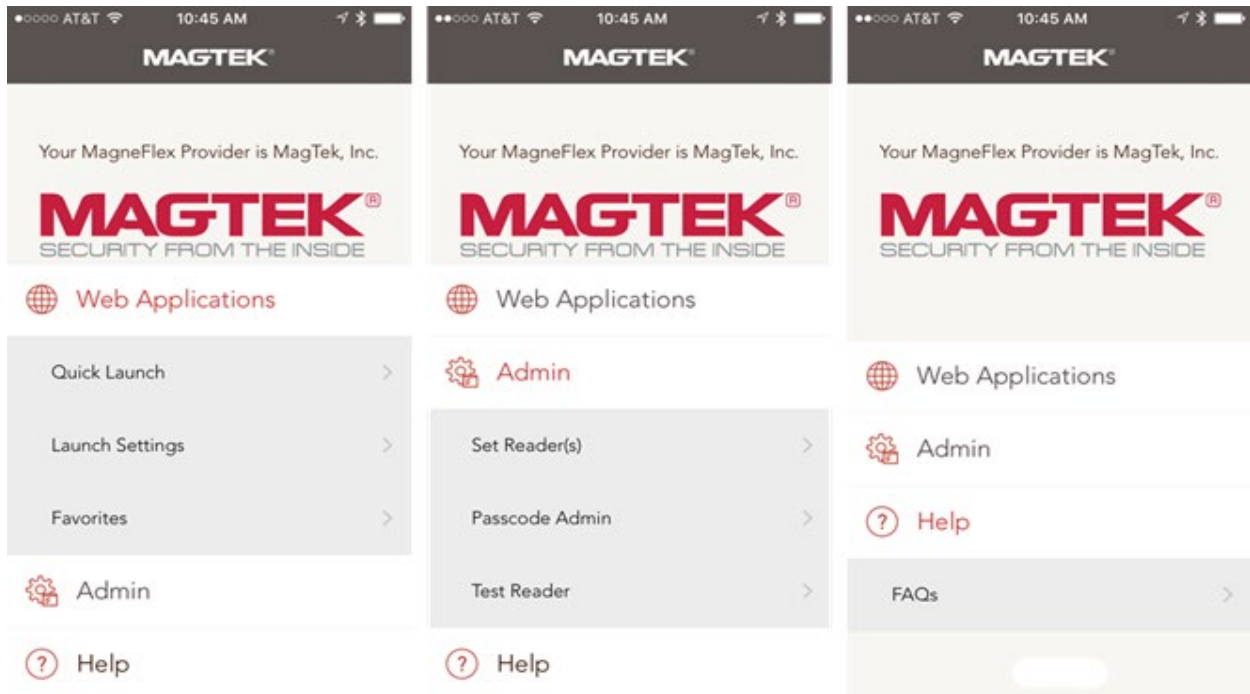


Figure H: Menu options, MagneFlex Browser for iOS

3.2.1 Web Applications

3.2.1.1 Quick Launch

Perform the action stored in **Launch Settings**.

3.2.1.2 Launch Settings (figure 3.3)

Determines the behavior of the application upon launch:

- **Default Web App:** Immediately surf the website stored in **Favorites** that has been marked as “Default”
- **List of Web Apps:** Open the list of websites stored in **Favorites**.
- **Blank Page:** Opens a blank page.
- **None:** Begins at the top menu.

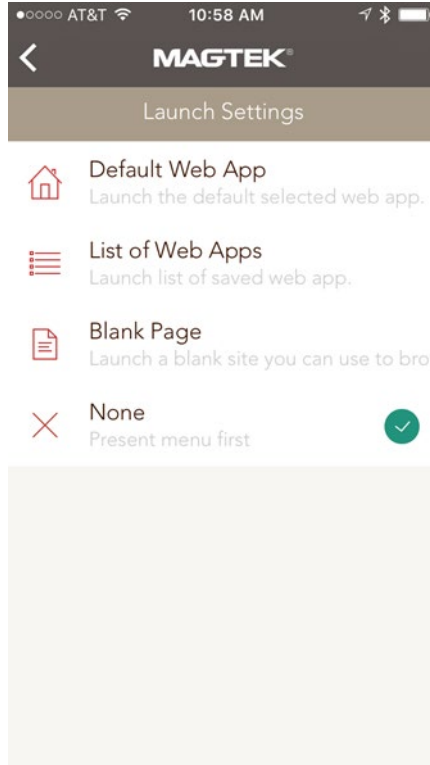



Figure I: Launch Settings, MagneFlex Browser for iOS

3.2.1.3 Favorites

A list of website URLs stored in the application. Allows the user to enter and store multiple URLs, as well as select websites to open.

The **Favorites** screen lists all websites that have been setup in the application. The user may launch websites from this screen, or they may add new favorites by selecting  **Add New Site**.

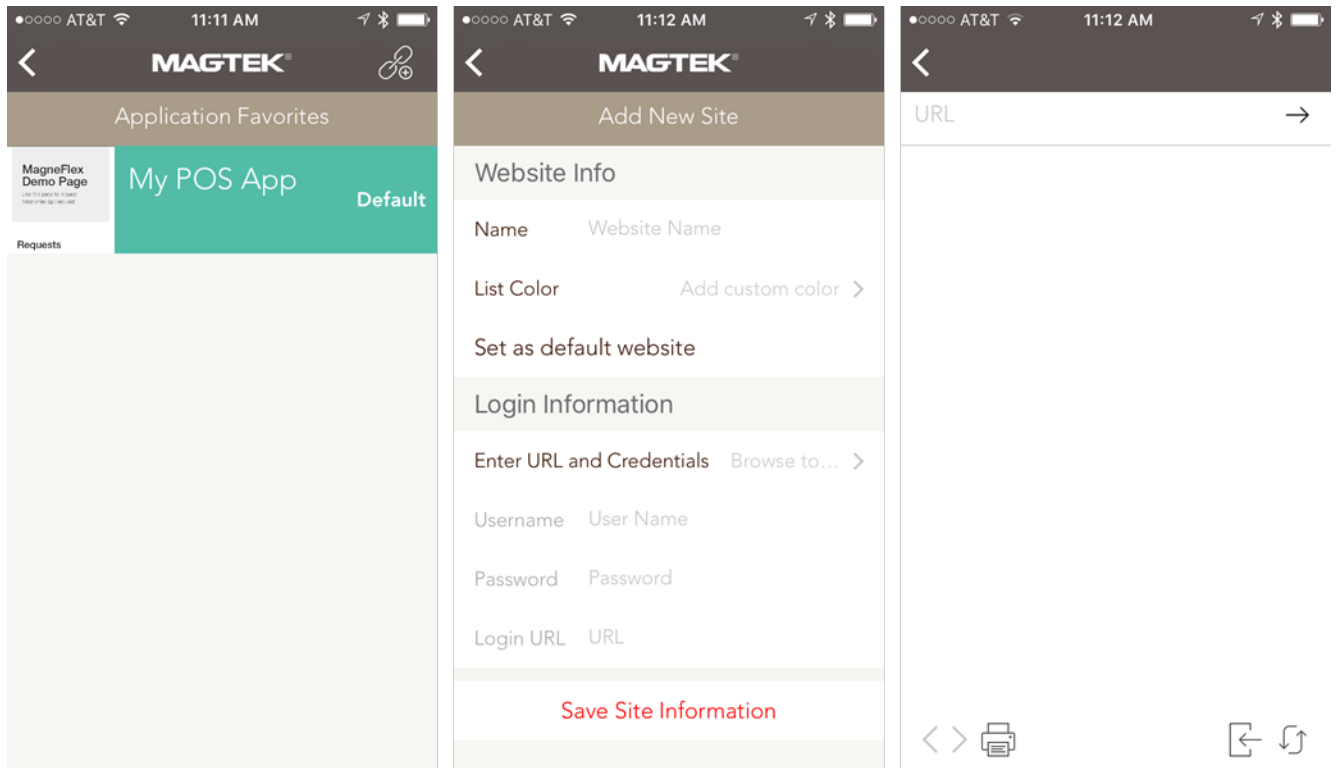



Figure J: Favorites menus, MagneFlex Browser for iOS

Add New Site: Add a new website favorite. Select **Save Site Information** once complete.

- **Name:** A “friendly” name for the website. Will be displayed on the **Favorites** list (example: “My POS App”).
- **List Color:** The background color of the website listing in **Favorites**.
- **Set as default website:** A toggle that sets this website as the default for **Launch Settings**.

Enter URL and Credentials: Selecting this item will cause the Browser to launch and display a blank page. Enter a URL and surf to the desired start point for the favorite. After completion select , and you will return to **Add New Site**. The start URL will show in **Login URL**. If the start page of the desired favorite requires *username* and *password* entry, the application will automatically detect the entry of these credentials and ask if you wish to store them. Selecting “yes” will cause the application to automatically log in the user every time this favorite is launched.

3.2.2 Admin

3.2.2.1 Set Readers(s)

Configure the application to communicate with various MagTek Readers.

The **Device Configuration** screen allows the user to select and configure the MagTek Readers to be used with **MagneFlex Browser**.

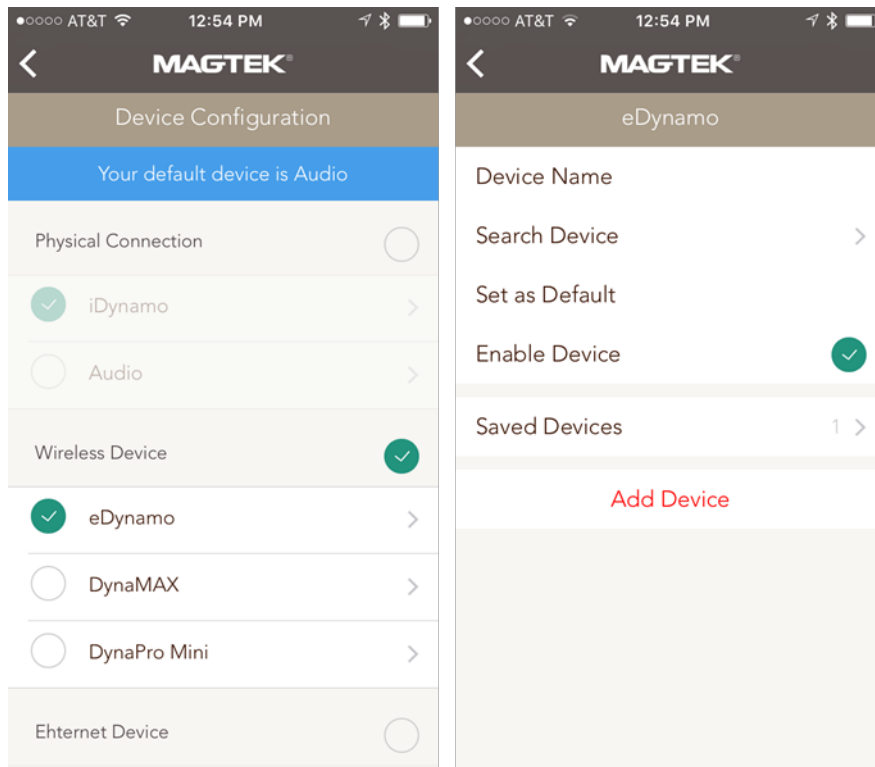


Figure K: Device Configuration, MagneFlex Browser for iOS

First select an interface type, and then select the Reader model. On the following screen, attributes for the Reader may be selected and entered.

- **Device Name:** Enter a name for the Reader (DynaPro/Go Ethernet/Wi-Fi only).
- **Device Address:** Enter the IP address of the Reader. (DynaPro/Go Ethernet/Wi-Fi only).
- **Search Device:** If the Reader is discoverable over Bluetooth, this option will be available. Select it to find Bluetooth Readers in range.
- **Set as Default:** Select if you wish to make this Reader the default Reader for the application.
- **Enable Device:** This selection shows whether the Reader is enabled for use.
- **Saved Devices:** Readers that have already been configured will be displayed in this list.

Add Device: This selection saves the Reader configuration and makes it available to the application for use.

3.2.2.2 Passcode Admin

Set a passcode lock for the application

3.2.2.3 Test Readers

Test a configured Reader for proper operation.

Magensa Services Developer Tool - MagneFlex Browser

The user may attach or associate any Reader type shown on this list to their mobile device and test its function by selecting the appropriate item.

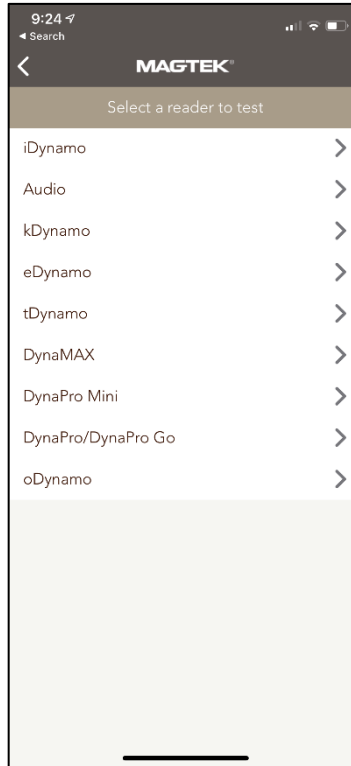


Figure L: Test Readers, MagneFlex Browser for iOS

The application will attempt to communicate with the Reader, and if successful, ask the user to swipe a card. The raw data from the swipe will be displayed on the screen. The Reader **DOES NOT** need to be configured on the **Device Configuration** screen first.

3.2.3 Help

- **FAQs:** Browse MagTek support website.

3.3 Android

MagneFlex Browser is available in Google Play. Search “MagneFlex”. This application is also available for customization and can be deployed to Google Play under the HTML Application developer’s own Google product plan. Please see a MagTek sales representative for further details.

The main menu for the Android version of the MagneFlex Browser can be accessed by tapping the left arrow icon (←) in the upper left corner of the screen.

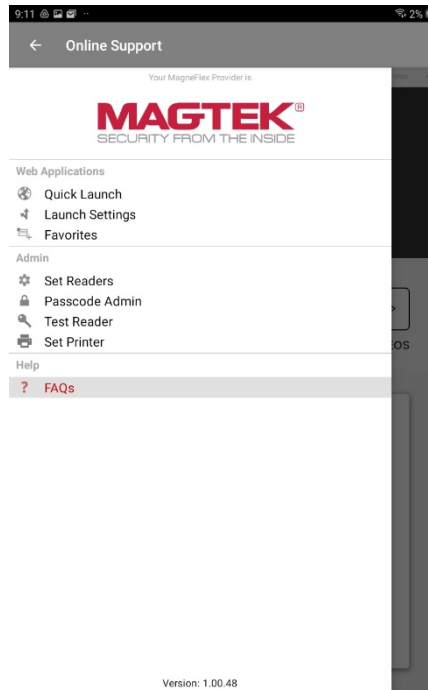


Figure M: Main Menu, MagneFlex Browser for Android

The following sections describe the options available from this menu.

3.3.1 Web Applications

3.3.1.1 Quick Launch

Perform the action stored in Launch Settings.

3.3.1.2 Launch Settings (figure 3.3)

Determines the behavior of the application upon launch:

- **Default Web App:** Immediately surf the website stored in **Favorites** that has been marked as “Default”
- **List of Web Apps:** Open the list of websites stored in **Favorites**.
- **Blank Page:** Opens a blank page.
- **None:** Begins at the top menu.

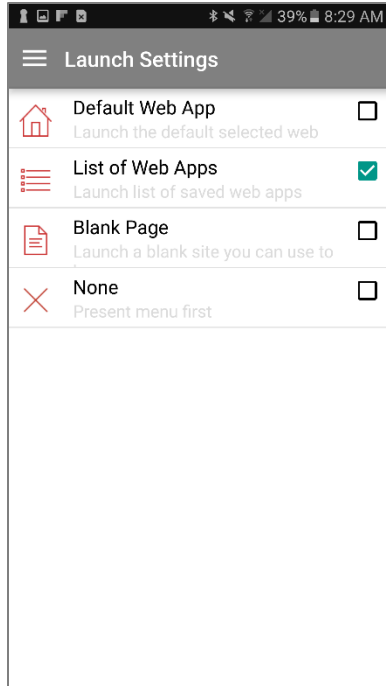



Figure N: Launch Settings, MagneFlex Browser for Android

3.3.1.3 Favorites

A list of website URLs stored in the application. Allows the user to enter and store multiple URLs, as well as select websites to open.

The **Favorites** screen lists all websites that have been setup in the application. The user may launch websites from this screen, or they may add new favorites by selecting  **Add New Site**.

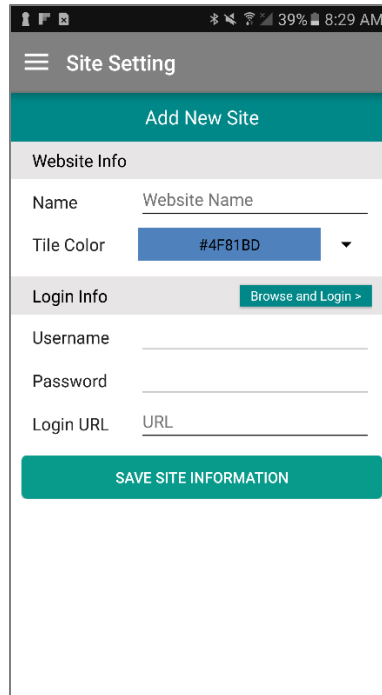



Figure O: Adding a favorite location, MagneFlex Browser for Android

Add New Site: Add a new website favorite. Select **Save Site Information** once complete.

- **Name:** A “friendly” name for the website. Will be displayed on the **Favorites** list (example: “My POS App”).
- **List Color:** The background color of the website listing in **Favorites**.
- **Set as default website:** A toggle that sets this website as the default for **Launch Settings**.

Enter URL and Credentials: Selecting this item will cause the Browser to launch and display a blank page. Enter a URL and surf to the desired start point for the favorite. After completion select , and you will return to **Add New Site**. The start URL will show in **Login URL**. If the start page of the desired favorite requires *username* and *password* entry, the application will automatically detect the entry of these credentials and ask if you wish to store them. Selecting “yes” will cause the application to automatically log in the user every time this favorite is launched.

3.3.2 Admin

3.3.2.1 Set Readers(s)

Configure the application to communicate with various MagTek Readers.

The **Device Configuration** screen allows the user to select and configure the MagTek Readers to be used with **MagneFlex Browser**.

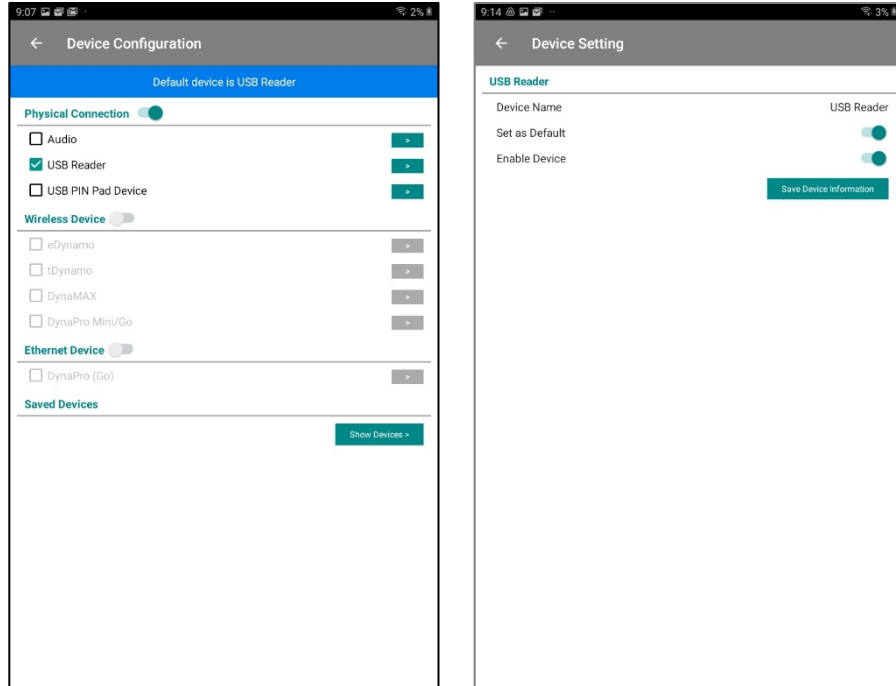


Figure P Device Configuration, MagneFlex Browser for Android

First select an interface type/type of connection, and then select the Reader model. On the following screen, attributes for the Reader may be selected and entered.

- **Device Name:** Enter a name for the Reader (DynaPro/Go Ethernet/Wi-Fi only). If the reader is a USB Reader such as iDynamo 6, then just select USB Reader and Save Device Information.
- **Device Address:** Enter the IP address of the Reader. (DynaPro/Go Ethernet/Wi-Fi only).
- **Search Device:** If the Reader is discoverable over Bluetooth, this option will be available. Select it to find Bluetooth Readers in range.
- **Set as Default:** Select if you wish to make this Reader the default Reader for the application.
- **Enable Device:** This selection shows whether the Reader is enabled for use.
- **Saved Devices:** Readers that have already been configured will be displayed in this list.
- **Add Device:** This selection saves the Reader configuration and makes it available to the application for use.

3.3.2.2 Passcode Admin

Set a passcode lock for the application.

3.3.2.3 Test Readers

Test a configured Reader for proper operation.

The user may attach or associate any Reader type shown on this list to their mobile device and test its function by selecting the appropriate item.

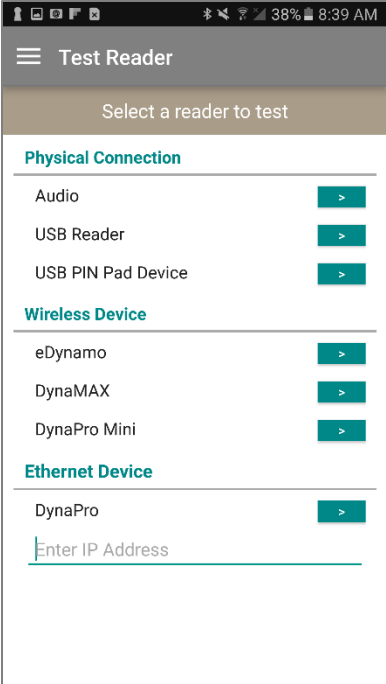
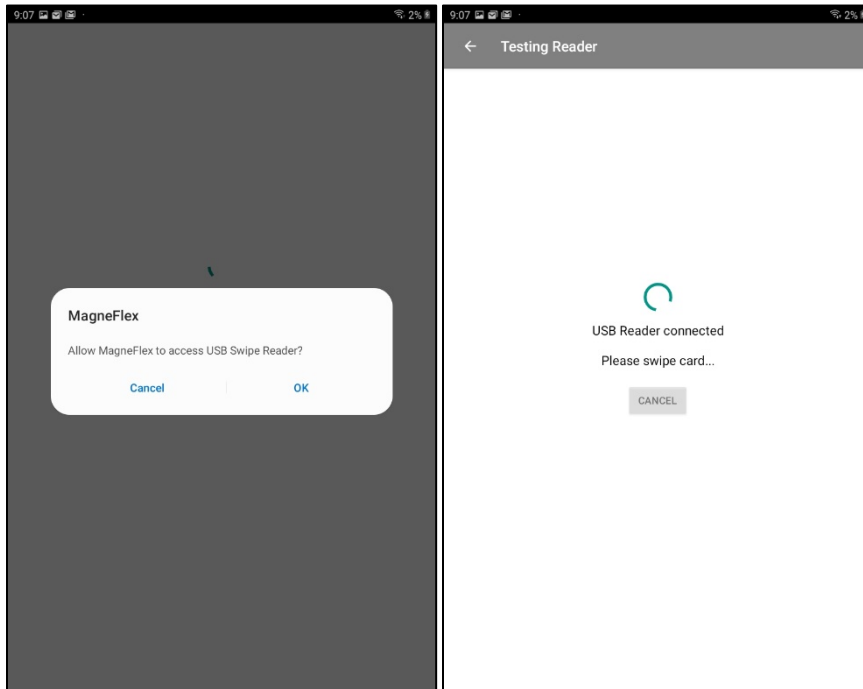


Figure Q: Test Reader, MagneFlex Browser for Android

The application will attempt to communicate with the Reader, and if successful, ask the user to swipe a card. The raw data from the swipe will be displayed on the screen. The Reader DOES NOT need to be configured on the **Device Configuration** screen first.

Magensa Services Developer Tool - MagneFlex Browser

When using a USB Reader, you will see a screen indicating that MagneFlex wants to access the USB Swipe Reader. You will need to press OK. Next, a window will open asking the user to “Please swipe card...”.



3.3.3 Help

- **FAQs:** Browse MagTek support website.

4 HTML Application Integration

MagneFlex iOS, Android, and MagneFlex v2 in Windows will use JavaScript request and callback. It is up to the client page to decide what to do with this information. Magensa provides a MagneFlex JS library ([MTMagneFlexLib.js](#)) that the customer can utilize.

MTMagneFlexLib will use, in this case `swipeCallBack` will be a function pointer for MagneFlex to callback on.

Example:

```
function requestCardSwipe()  
{  
  
    var readerParam = new MTMagneFlexParameter.readerParameter();  
    readerParam.timeLimit = '3C';  
  
    var requestParam = new MTMagneFlexParameter.requestParameter();  
    requestParam.operation = MagneFlexOp.REQUEST_CARD_SWIPE;  
    requestParam.httpMethod = HTTPMethod.NONE;  
    requestParam.customDisplayMessage = 'Please Swipe Card';  
    requestParam.fullScreenMode = false;  
    requestParam.closeDeviceAfter = true;  
    requestParam.destinationURL = '';  
  
    MTMagneFlexLib.requestCardSwipe(readerParam, requestParam,  
swipeCallBack);  
}
```

Parameter	Description
readerParameter	Reader parameters to be sent to the Reader. Ex) timeLimit, tone, option, maxPinLength, minPinLength, pinOption etc.
requestParameter	Parameters for Request to be sent Ex) Operation (refer to operation definition in next chapter), httpMethod (NONE), closeDeviceAfter

4.1 Operation Definitions and Inputs

MagneFlex Browser recognizes and performs the following operations:

4.1.1 requestCardSwipe

Commands the Reader to perform a magnetic stripe card swipe. There is no readerParameter with this operation. This operation is intended for our SCRA family of magnetic stripe reading Readers.

```
function requestCardSwipe()  
{  
  
    var readerParam = new MTMagneFlexParameter.readerParameter();  
    readerParam.timeLimit = '3C';  
  
    var requestParam = new MTMagneFlexParameter.requestParameter();  
    requestParam.operation = MagneFlexOp.REQUEST_CARD_SWIPE;  
    requestParam.httpMethod = HTTPMethod.NONE;  
    requestParam.customDisplayMessage = 'Please Swipe Card';  
    requestParam.closeDeviceAfter = true;  
  
    MTMagneFlexLib.requestCardSwipe(readerParam, requestParam,  
swipeCallBack);  
}
```

Parameter	Description
operation	REQUEST_CARD_SWIPE
httpMethod	NONE
customDisplayMessage	When modal prompting user, this message will be displayed to let user know why they card swipe is needed.
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.1.2 requestSendCommand

Commands the Reader to perform the operations defined in readerParameter. See the Programmer's Reference Manual for the Reader for a list of available commands and the byte sequence required.

```
function requestSendCommand(command)  
{  
    var readerParam = new MTMagneFlexParameter.readerParameter();  
    readerParam.timeLimit = '3C';  
    readerParam.waitForReport = '01';  
    readerParam.command = command;  
    readerParam.endSession = true;  
    readerParam.commandType = 'GET';  
  
    var requestParam = new MTMagneFlexParameter.requestParameter();  
    requestParam.operation = MagneFlexOp.REQUEST_SEND_COMMAND;  
    requestParam.httpMethod = HTTPMethod.NONE;
```

Magensa Services Developer Tool - MagneFlex Browser

```
requestParam.fullScreenMode = false;
requestParam.closeDeviceAfter = true;
MTMagneFlexLib.requestSendCommand(readerParam, requestParam,
commandCallBack);
}
```

Parameter	Description
command	Command to send to device.
endSession	Lets DynaPro device family know if endsession is needed.
commandType	GET/SET, only applies to DynaPro Devices.
operation	REQUEST_SEND_COMMAND
httpMethod	NONE
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.1.3 requestSendExtendedCommand

Commands the Reader to perform the extended command defined in readerParameter. See the Programmer's Reference Manual for the Reader for a list of extended commands and the byte sequence required.

```
function requestSendExtendedCommand(extendedCommand)
{
    var readerParam = new MTMagneFlexParameter.readerParameter();
    readerParam.timeLimit = '3C';
    readerParam.commandValue = extendedCommand;

    var requestParam = new MTMagneFlexParameter.requestParameter();
    requestParam.operation = MagneFlexOp.REQUEST_SEND_EXTENDED_COMMAND;
    requestParam.httpMethod = HTTPMethod.NONE;
    requestParam.fullScreenMode = false;
    requestParam.closeDeviceAfter = true;
    MTMagneFlexLib.requestSendCommand(readerParam, requestParam,
commandCallBack);
}
```

Parameter	Description
commandValue	Extended command to send.
operation	REQUEST_SEND_EXTENDED_COMMAND
httpMethod	NONE
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.1.4 requestStartEMVTransaction

Commands the Reader to perform an EMV transaction (if capable). This operation must be accompanied by readerParameter. See the Reader's Programmer's Reference Manual, "Start EMV Transaction".

```
function requestSmartCard(quickChip) {  
  
    var readerParam = new MTMagneFlexParameter.readerParameter();  
    readerParam.timeLimit = '3C';  
    readerParam.endSession = "true";  
    readerParam.amount = "99.99";  
    readerParam.cashBack = "9.99";  
    readerParam.option = "00";  
    readerParam.currencyCode = "0840";  
    readerParam.cardType = "02";  
    readerParam.transactionType = "00";  
    if (quickChip)  
        readerParam.quickChipMode = "true";  
    else  
        readerParam.quickChipMode = "false";  
    readerParam.sendBatchData = "true";  
  
    var requestParam = new MTMagneFlexParameter.requestParameter();  
    requestParam.operation = MagneFlexOp.REQUEST_SMART_CARD_TRANSACTION;  
    requestParam.httpMethod = HTTPMethod.NONE;  
    requestParam.fullScreenMode = false;  
    requestParam.closeDeviceAfter = false;  
    MTMagneFlexLib.requestSmartCard(readerParam, requestParam,  
arcqDataCallBack, batchDataCallback);  
}
```

readerParameter is dependent on the type of Reader used. Please refer to COMMANDS documentation.

Parameter	Description
amount	The amount to be used and authorized in decimal format. 1.01 = 1 dollar and 1 cent
cashback	The amount to be used and authorized in decimal format. 1.01 = 1 dollar and 1 cent
cardType	01 : Magnetic Strip 02 : Contact Smart Card 03 : Magnetic stripe or contact smart card 04 : Contactless smart card (Not supported on DynaPro Mini) 05 : Contactless smart card + magnetic stripe 06 : Contactless smart card + contact smart card 07 : Magnetic stripe + contact smart card + contactless smart card
quickChipMode	QuickChip enabling true or false.
operation	REQUEST_SMART_CARD_TRANSACTION
httpMethod	NONE

Magensa Services Developer Tool - MagneFlex Browser

Parameter	Description
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.1.5 sendARPC

Commands the Reader to accept an ARPC returned by the card issuer at the completion of an EMV transaction (if required). This operation must be accompanied by readerParameter containing the ARPC data. See the Reader's Programmer's Reference Manual, "Acquirer Response ARPC".

```
function sendARPC(data) {
    var readerParam = new MTMagneFlexParameter.readerParameter();
    readerParam.commandValue = data;

    var requestParam = new MTMagneFlexParameter.requestParameter();
    requestParam.operation = MagneFlexOp.REQUEST_SEND_ARPC;
    requestParam.httpMethod = HTTPMethod.NONE;
    requestParam.fullScreenMode = false;
    requestParam.closeDeviceAfter = true;

    MTMagneFlexLib.requestSendARPC(readerParam, requestParam);
}
```

Parameter	Description
commandValue	Acquirer response to card
operation	REQUEST_SEND_ARPC
httpMethod	NONE
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.1.6 requestPIN

Commands the Reader, if it has a PIN PAD, to prompt the user for the entry of a PIN. This operation must be accompanied by readerParameter. See the Reader's Programmer's Reference Manual, "Request PIN Entry".

```
function requestPINEntry()
{
    var readerParam = new MTMagneFlexParameter.readerParameter();
    readerParam.endSession = true;
    readerParam.timeLimit = "3C";
    readerParam.fieldSeparator = "|";
    readerParam.maxPinLength = "0C";
    readerParam.minPinLength = "04";
    readerParam.pinMode = "0";
    readerParam.tone = "01";
}
```

Magensa Services Developer Tool - MagneFlex Browser

```
readerParam.pinOption = "02";

var requestParam = new MTMagneFlexParameter.requestParameter();
requestParam.operation = MagneFlexOp.REQUEST_PIN_ENTRY;
requestParam.httpMethod = HTTPMethod.NONE;
requestParam.fullScreenMode = false;
requestParam.closeDeviceAfter = true;
MTMagneFlexLib.requestPinEntry(readerParam, requestParam,
pinDataCallback);
}
readerParameter : please refer to DynaPro command manual.
```

Parameter	Description
pinMode	Message to display as a user prompt: 0 : ENTER PIN 3 : Incorrect PIN Please re-enter PIN 4 : FOR VERIFICATION RE-ENTER PIN
Tone	Tones to use: 0 = No sound 1 = One beep 2 = Two beep
maxPinLength	Maximum PIN length. Must be less than 13
minPinLength	Minimum PIN length. Must be greater than 3.
operation	REQUEST_PIN_ENTRY
httpMethod	NONE
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.1.7 requestManualCardEntry

Commands the Reader, if it has a PIN PAD, to prompt the user for the entry of a card number. This operation must be accompanied by readerParameter. See the Reader's Programmer's Reference Manual, "Request Manual Card Entry".

```
function requestManualEntry()
{
    var readerParam = new MTMagneFlexParameter.readerParameter();
    readerParam.endSession = true;
    readerParam.timeLimit = "3C";
    readerParam.tone = "01";
    readerParam.option = "08";

    var requestParam = new MTMagneFlexParameter.requestParameter();
    requestParam.operation = MagneFlexOp.REQUEST_MANUAL_CARD_ENTRY;
    requestParam.httpMethod = HTTPMethod.NONE;
    requestParam.fullScreenMode = false;
    requestParam.closeDeviceAfter = true;
```

Magensa Services Developer Tool - MagneFlex Browser

```
MTMagneFlexLib.requestManualEntry(readerParam, requestParam,  
manualCardDataCallback);
```

```
}
```

Parameter	Description
tone	Set to 01
operation	REQUEST_MANUAL_CARD_ENTRY
httpMethod	NONE
closeDeviceAfter	Lets MagneFlex know to close device after operation is done or not.

4.2 Operation Outputs

MagneFlex Browser will return to the HTML Application data returned by the Reader using JavaScript callback provided.

Variable	Description
opType	Indicates the operation performed: requestCardSwipe: CARD_SWIPE_RESULT requestSendCommand: SEND_COMMAND_RESULT requestSendExtendedCommand: SEND_EXTENDED_COMMAND_RESULT requestStartEMVTransaction: EMV_ARQC_RESULT sendARPC: EMV_TRANS_RESULT requestPIN: PIN_RESULT requestManualCard: CARD_MANUAL_SWIPE requestSignature: SIGNATURE_RESULT
cardDataIn	requestCardSwipe and requestManualCard only. The result of these operations in TLV format. See appendix B for common TLV tag definitions.
commandID	requestSendCommand and requestSendExtendedCommand only. An echo of the command that was sent in the operation.
commandResult	requestSendCommand and requestSendExtendedCommand only. Data returned from the Reader as a result of the command.
arqcData	requestStartEMVTransaction only. The ARQC data from an EMV transaction.
batchData	sendARPC only. The batch data from an EMV transaction.
operationStatus	requestPIN only. The status of the PIN operation from the Reader.

Magensa Services Developer Tool - MagneFlex Browser

Variable	Description
pinEPB	requestPIN only. The encrypted PIN block from the PIN operation.
pinKSN	requestPIN only. The KSN of the encrypted PIN block from the PIN operation.
signatureDataIn	requestSignature only. Signature data from the Reader.
signatureResult	requestSignature only. Result of the signature operation.
passInData	Echo of the passInData variable in the HTML link.

5 Test page

As part of our developer tools, we also include a test demo page that developers can access:

<https://demo.magensa.net/magneflex/>

It includes MSR, PINpad, MICR, Command Builder, and Misc. Request:

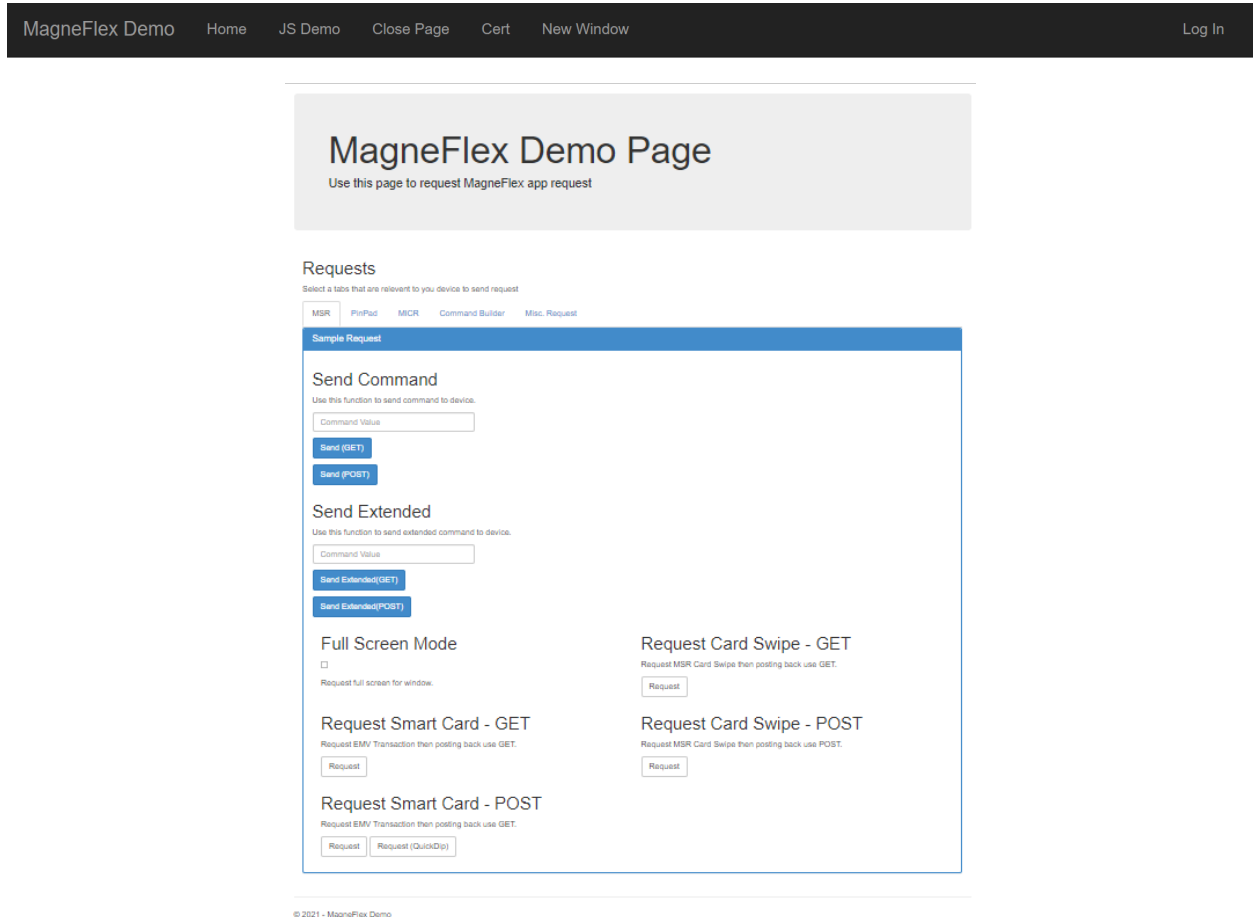


Figure R: MagneFlex Browser Demo Page

6 Appendix A – Supported Readers

MagTek Devices	Windows	Apple iOS	Android
MagTek SCRA	USB	NA	NA
DynaWave	USB & UART	NA	USB
iDynamo 5	NA	Lightning	NA
iDynamo 6	USB	Lightning	USB
kDynamo	NA	Lightning	USB
tDynamo	USB & Bluetooth	Bluetooth	Bluetooth & USB
uDynamo	USB	Audio jack	Audio jack
cDynamo	NA	Lightning	NA
eDynamo	USB	Bluetooth	Bluetooth & USB
DynaPro	USB & Ethernet	Ethernet	USB & Ethernet
DynaPro Go	USB & Wi-Fi	Wi-Fi	Bluetooth & Wi-Fi & USB
DynaPro mini	USB	Bluetooth	Bluetooth & USB
DynaPAD	USB	NA	NA
Dynamag	USB	NA	NA

7 Appendix B – Common MagTek Magnetic Stripe Reader TLV Tags

DFDF25	Device Serial Number
DFDF30	Masked T1 Status
DFDF31	Masked T1
DFDF32	Masked T2 Status
DFDF33	Masked T2
DFDF34	Masked T3 Status
DFDF35	Masked T3
DFDF36	Encrypted T1 Status
DFDF37	Encrypted T1
DFDF38	Encrypted T2 Status
DFDF39	Encrypted T2
DFDF3A	Encrypted T3 Status
DFDF3B	Encrypted T3
DFDF3C	Encrypted MagnePrint
DFDF3D	MS2.0 Status
DFDF43	MagnePrint Status Data
DFDF50	Key Serial Number