VICTORIAN PLAYER ACCOUNT EQUIPMENT TECHNICAL REQUIREMENTS DOCUMENT

December 2016

Version 2.0

Ref: CD/16/33570

Victorian Commission for Gambling and Liquor Regulation

State Government Victoria

TABLE OF CONTENTS

1 Glossary

This chapter sets out the glossary of standard terms and abbreviations used by the Commission and relevant to the Player Account Equipment Technical Requirements document.

Term or Abbreviation	Description
Act	Means the *Gambling Regulation Act 2003 (Vic)*, as amended from time to time
Ancillary Service System	A system, including software and hardware that caters for provision of additional services offered within a venue that may be connected to gaming and/or monitoring and/or PCS equipment.
Australian/New Zealand Gaming Machine National Standards	Refer to the National Standards
Casino	Has the same meaning as in the Casino Control Act
Casino Control Act	Means the *Casino Control Act 1991* as amended from time to time
Casino Operator	Has the same meaning as defined in the Casino Control Act
CMCS	The Central Monitoring and Control System, made up of host CMCS, Venue CMCS and network components, of the Licensee's gaming monitoring network as refer to in section 3.1.6 of the VCR,
Commission	The Victorian Commission for Gambling and Liquor Regulation established under the *Victorian Commission for Liquor and Gambling Regulation Act* 2011 or any successor body
Commission Standards	The relevant Commission gaming standards are the standards referred to in section 3.5.3 of the Act
Data	Means all data and expressions of data contained in, or processed or generated by, the Pre-commitment System including without limitation: i). All data and expressions of data comprising reports generated by the Pre-commitment System; and ii). All data and expressions of data about or relating to or generated by agents and contractors stored within the Pre-commitment System.
Gambling Regulation (Pre-commitment and Loyalty) Regulations.	Regulations that are made under section 11.2.1 of the *Gambling Regulation Act 2003*, as amended from time to time.
Gaming Machine	Has the same meaning as defined in the Act
Gaming Monitoring Activities	Means the establishment, operation and maintenance of the Monitoring System, the provision of Monitoring Services and the sale, supply and possession of Monitoring Equipment in accordance with section 3.4.4(1)(a), (b) and (c) of the Act and the Scope of Services set out in the Monitoring Licence and Related Agreements

Term or Abbreviation	Description
Gaming Networks	The networks that make up gaming include the systems and components of gaming, monitoring and pre-commitment networks
Gaming Venue	Has the same meaning as Venue
Hardware	All physical components (electrical and mechanical) making up the Player Account Equipment
Interface Card	A computer device which is located inside gaming equipment, or a Gaming Machine, and performs various functions such as protocol conversion, for example a Slot Machine Interface Board in a gaming machine
Monitoring Licence	Means the licence granted and issued under the Act by the Minister to authorise the conduct of the gaming monitoring activities
Monitoring Licensee	The holder of the licence granted and issued under the Act by the Minister to authorise the conduct of the gaming monitoring activities
Minister	The Victorian Minister responsible for Liquor and Gaming Regulation
Monitoring Equipment	Has the same meaning as defined in the Act
Monitoring System	Means the electronic monitoring system referred to in section 3.4.4 of the Act and includes, without limitation, all adaptations, modifications, enhancements to that system made at any time before or during the term
National Standards	The core requirements, common to all jurisdictions, for the design of Gaming Machines and games for operation throughout Australia and New Zealand and to guide to suppliers of testing services in their testing for compliance with the standard
Pre-commitment Scheme	Pre-commitment scheme means the interface between player account equipment and the player cards with the pre-commitment system that allows players to provide information to and receive information from the pre-commitment system
PIN	Personal Identification Number
Player Account Equipment (PAE)	Has the same meaning as defined in the section 3.8A.1 of the Act
Pre-commitment System (PCS)	Means the electronic pre-commitment system referred to in sections 3.4.4 (1)(a), (b) and (c) of the Act
Regulations	See: Gambling Regulation (Pre-commitment and Loyalty) Regulations
Roll of Manufacturers, Suppliers and Testers	Has the same meaning as The Roll set out in section 3.4.60 of the Act
Tester	Means a supplier of testing services listed on the Roll of Manufacturers, Suppliers and Testers as described in the chapter 3 of the Act
Touch Screen	A display that can interact with the user by touching the video monitor screen
Unique Identification Number	Means the number allocated by the pre-commitment system and stored on a person's player card that enables the pre-commitment system to identify the persons player account
VCGLR	The Victorian Commission for Gambling and Liquor Regulation

Term or Abbreviation	Description
VCR	Victorian Central Monitoring and Control System Requirements
Venue CMCS	Components of the CMCS located within a Venue
Venue Operator	The holder of a Venue Operator's Licence, a Licence issued under Division 2 of Part 4 of Chapter 3 of the Act, as defined in the Act
Venue	Has the same meaning as an approved venue as defined in the Act, as well as the Melbourne Casino.
Victorian Technical Standards	Means the current versions of the: ▪ Victorian Pre-commitment System Requirements document issued by the Commission, as amended by the Commission from time to time ▪ Victorian Player Account Equipment Technical Requirements document issued by the Commission, as amended by the Commission from time to time (this document) ▪ Victorian Central Monitoring and Control System Requirements document issued by the Commission, as amended by the Commission from time to time ▪ Australia/New Zealand Gaming Machine National Standard (National Standard) as amended from time to time and ▪ Victorian Appendix to the Australia/New Zealand Gaming Machine National Standard (Victorian Appendix), as amended by the Commission from time to time

2 Introduction

This chapter introduces the background to the Victorian Player Account Equipment Technical Requirements document.

2.1 Document purpose

The document sets the minimum specifications of the Player Account Equipment (PAE) and identifies high-level technical requirements that PAE must meet for the operation of the pre-commitment scheme.

The standard does not prescribe the make, model or software interface of or for PAE, and are considered out of scope for the purpose of this document. Appendix D includes the Intralot PAE Interface Specification required for PAE to connect into the system.

This standard is made in accordance with section 10.1.5A of the Act.

The contents of this document may be updated from time to time.

2.2 Background

Pre-commitment is a technological system that helps to minimise harm by providing a tool to assist players to control their gambling behaviour and avoid escalating gaming into harmful levels of play. The Victorian Government's pre-commitment policy states that the pre-commitment scheme:

- is mandatory on all gaming machines, at all gaming venues across Victoria including the Melbourne casino
- is provided via a networked system with gaming machines connected to a central database
- shares existing infrastructure to ensure maximum efficiency and economies of scale
- shares PAE with loyalty systems to ensure maximum efficiency and economies of scale

The installation of PAE on or in a gaming machine and/or in a gaming venue is required to facilitate the pre-commitment scheme. It is a suite of hardware devices that each gaming venue will be required by legislation to procure and install before the pre-commitment scheme commences on 1 December 2015. The description of the required PAE is detailed in the Gambling Regulation (Pre-commitment and Loyalty) Regulations 2014, as amended from time to time.

In the Act, Player Account Equipment means the following equipment:

- interactive display at the gaming machine
- card reader at the gaming machine
- kiosk
- card reader at the kiosk
- keypad at the player service point
- card reader and card encoder at the player service point
- player card
- .or any other equipment that is prescribed as player account equipment

2.3 Related documents

The technical requirements described in this document align with the following Australian and Victorian instruments, as amended from time to time:

- Australian/New Zealand Gaming Machine National Standard as approved in Victoria
- Victorian Appendix to the Australian/New Zealand Gaming Machine National Standard
- *Gambling Regulation Act 2003*
- *Casino Control Act 1991*
- Gambling Regulation (Pre-commitment and Loyalty Scheme) Regulations 2014
- Victorian Pre-commitment System Requirements document

The technical requirements described in this document refer to the following ISO standards:

- ISO/IEC 7811 - Identification cards — Recording technique
- ISO/IEC 7810 - Identification cards — Physical characteristics

2.4 Document scope

The player account equipment for the purpose of this document includes the following equipment:

- interactive display at the gaming machine
- card reader at the gaming machine
- kiosk
- card reader at the kiosk
- service point workstation
- keypad at the player service point
- card reader and card encoder at the player service point
- player card

3 General requirements

This chapter describes the overarching requirements relating to all PAE in Victoria.

3.1 General

PAE must connect to the pre-commitment system. This connection must be fit for purpose and compatible with the approved PCS.

Req ID	Technical Requirements
3.1.1	PAE must connect to and interact with the Pre-commitment System.
3.1.2	PAE must meet the requirements of the Act, Regulations and Standards.
3.1.3	PAE must not impede or affect the integrity or conduct of gaming and monitoring.
3.1.4	PAE must function in the manner in which it is designed and programmed to function

4 Regulatory requirements

This chapter sets out the regulatory requirements for Player Account Equipment that must be followed in Victoria.

4.1 General

Req ID	Technical Requirements
4.1.1	A variation to a gaming machine type must be approved by the Commission for PAE to be installed on or in a gaming machine. This approval must be obtained prior to the PAE being installed on or in a gaming machine.
4.1.2	Installation of PAE at the gaming machine must meet the requirements set down in section 3.4.5 (c)(ia) and section 3.5.5 of the Act or meet requirements set down in the *Casino Control Act*.
4.1.3	PAE installed on or in a gaming machine must be certified, in a form approved by the Commission that the equipment is functioning in the manner in which it is designed and programmed to function to meet the conditions of section 3.8A.7(2) of the Act.

4.2 PAE requirements

The intent of this section is to describe the equipment requirements for PAE. The technical minimum specifications for each component are detailed in *Appendix A and Appendix D*.

In accordance with Section 3.5.36D (2)(c) of the Act if an approved venue is operating a loyalty scheme then all PAE must be used for the purposes of the pre-commitment scheme and the loyalty scheme.

4.2.1 Kiosk

Req ID	Technical Requirements
4.2.1.1	The kiosk must be able to be secured to a fixed location.
4.2.1.2	The kiosk may be wall mounted, freestanding, or countertop mounted
4.2.1.3	The kiosk must have a card reader integrated into the configuration, which interacts with the pre-commitment system website.
4.2.1.4	The kiosk must provide controlled access to the pre-commitment public kiosk website via a graphical user interface that restricts players and members of the general public's access to the menu options available only within the graphical user interface. Administrative access must not be accessible to players or members of the general public.
4.2.1.5	The kiosk must comply with the Pre-Commitment Kiosk implementation style guide, as attached as Appendix E.

4.2.2 Keypad

Req ID	Technical Requirements
4.2.2.1	The keypad must be suitable for entry of a numeric PIN.
4.2.2.2	The keypad must be either a physical (device) or a display that supports a virtual keypad.
4.2.2.3	The keypad must interact with the pre-commitment system website.

4.2.3 Card encoder and or reader

Req ID	Technical Requirements
4.2.3.1	The card reader and or encoder must interact with the pre-commitment website.
4.2.3.2	The card reader and/or encoder must have the capability to read and/or write to at least track 2 of a magnetic stripe card for the purposes of pre-commitment.
4.2.3.3	The card encoder must encode data on player cards that complies with ISO/IEC 7811.
4.2.3.4	The encoder must be capable of writing up to and including at least position 22 of Track 2. *For further details refer to Appendix B – Player Card Pre-commitment Configuration*
4.2.3.5	Positions 12-20 of Track 2 of the magnetic stripe card must be used for pre-commitment scheme player's unique identifier. Note: *Positions 1-11 of Track 2 of the magnetic stripe card may be used for identifiers relevant to other Ancillary Service Systems, e.g. loyalty schemes.* *For further details refer to Appendix B – Player Card Pre-commitment Configuration*

4.2.3.A Player cards (encoded)

Req ID	Technical Requirements
4.2.3.A.1	The pre-commitment registered player's unique identification number must be encoded on the magnetic stripe card. Note: *The magnetic stripe card may also be encoded to contain the membership identifier of a player participating in a loyalty scheme.*
4.2.3.A.2	If a player is registered for pre-commitment and participates in a loyalty scheme then it is mandatory that the two identifiers are encoded to co-exist on the same magnetic stripe card.
4.2.3.A.4	The magnetic stripe on the card, at a minimum, must support low coercivity as defined in ISO/IEC 7811.
4.2.3.A.5	The unique identification number must be encoded as numeric ASCII characters. *For further details refer to Appendix B – Player Card Pre-commitment Configuration*

4.2.4 The PAE to be located in each gaming machine includes:

4.2.4.A Card reader

Req ID	Technical Requirements
4.2.4.A.1	The card reader must be capable of insertion and reading of an encoded magnetic stripe card.
4.2.4.A.2	The gaming machine card reader must be connected either directly to the monitoring Slot Machine Interface Board (SMIB) or via an ancillary service interface board that is connected directly to the monitoring SMIB.
4.2.4.A.3	The gaming machine card reader must be capable of reading up to and including position 22 of Track 2 of the encoded card.
4.2.4.A.4	The gaming machine card reader must interact with the pre-commitment system.

For further details refer to Appendix A – Minimum specifications for PAE and Appendix B – Player Card Pre-Commitment Configuration

4.2.4.B Interactive display

Req ID	Technical Requirements
4.2.4.B.1	The interactive display must be capable of displaying Half Video Graphics Array (HVGA) resolution video images, and capturing player interaction by touch.
4.2.4.B.2	The interactive display must either be directly connected to the Monitoring SMIB for both touch interactivity and graphics display delivery or via an ancillary service interface board that is connected directly to the Monitoring SMIB.
4.2.4.B.3	The interactive display must interact with the pre-commitment system. Where the interactive display is shared with other ancillary services such as loyalty, and a player has an active pre-commitment session, the ancillary service must provide a button on the interactive display to allow a player to return to the pre-commitment system on demand.
4.2.4.B.4	Where the interactive display is used for other purposes than pre-commitment, pre-commitment information and messages must be displayed without delay.
4.2.4.B.5	Where the interactive display is used for other purposes than pre-commitment, pre-commitment information or messages must not be interrupted or overwritten by other ancillary services.
	For further details refer to Appendix A – Minimum specifications for PAE

4.2.5 PAE infrastructure sharing and security

Req ID	Technical Requirements
4.2.5.1	PAE configurations must enable the sharing of infrastructure between pre-commitment and loyalty scheme(s). See section 3.5.36D of the Act. Note: *Infrastructure includes but is not limited to interactive display screen, card readers/encoders and kiosks*.
4.2.5.2	The card reader/encoder and the keypad must be able to facilitate manual or automatic entry of a unique identification number, and, where PAE is shared, other numbers, e.g. a Loyalty ID number, for the purpose of using that data to encode a magnetic stripe card.
4.2.5.3	Pre-commitment data must pass through player account equipment in a secure manner. Pre-commitment data must not be stored or modified for any purpose other than pre-commitment

4.2.6 PAE interface certification

In accordance with section 3.8A.7 the Act, the functional integrity of PAE must be certified by a technician and in accordance with section 3.8A.12 of the Act, the operator must ensure ongoing functional integrity.

Req ID	Technical Requirements
4.2.6.1	PAE must connect to, be compatible with and interact with the pre-commitment system.
4.2.6.2	PAE installed in a venue must function in the manner in which it is designed and programmed to function.

For further details, refer to Appendix C – Certification & Approval Process for PAE

5 Equipment Support and Maintenance

This chapter sets out the hardware support requirements for PAE that must be followed for operations in Victoria.

5.1 Maintenance

Req ID	Technical Requirements
5.1.1	Maintenance and the issuance of certification of PAE installed on or in a gaming machine must be conducted by a technician who holds a gaming industry employee's licence that is contracted by the venue operator, or, for PAE located within the Casino, maintenance and the issuance of certification must be conducted by persons holding a licence issued under Part 4 of the *Casino Control Act*.
5.1.2	Maintenance of PAE must be carried out in such a way that it does not contravene the approval for gaming machines, monitoring system or pre-commitment system.
5.1.3	Maintenance or repair of PAE fitted to approved gaming machines must be undertaken using replacement parts that conform to the Commission approval(s).
5.1.4	Hardware maintenance of equipment must not involve: • Testing and fault diagnosis requiring the cutting, drilling or addition of electronic circuitry; • Thermal overstressing of components; or • Removal or insertion of components while power is applied to the equipment, unless the equipment has been specifically designed to withstand such actions and then only by following the appropriate procedures laid down by the manufacturers.
5.1.5	All hardware maintenance must follow industry best-practice with respect to protecting the equipment from static discharge and where appropriate, the following shall be adhered to: • All components and assemblies must be stored and transported in anti-static packaging at all times; • No components or assemblies are to be touched unless the technician is earthed via a wrist strap or other earthing device; and • Maintenance work-areas must be earthed and fitted with earthed floor mats, earthed bench mats and wrist strap earth points.

6 Approval and Certification Requirements

This chapter sets out the Commission's requirements for approval, certification and installation in Victoria. It applies to the supply and installation of Player Account Equipment.

6.1 Approvals and authorisation

In accordance with Sections 3.8A.7 of the Act the functional integrity of PAE at the gaming machine must be certified. The following are the Commissions requirements relating to installation, certification and, where necessary, approval of the PAE.

In approving a variation to a gaming machine type for the installation of player account equipment, the Commission may have regard to a recommendation from a Tester.

Req ID	Technical Requirements
6.1.1	Installation of any PAE on an approved gaming machine must be carried out by holders of the gaming industry employees licence that are: • employed by an entity on the Roll of Manufacturers, Suppliers and Testers, and • contracted by the Venue Operator; or • hold a licence issued under Part 4 of the *Casino Control Act.*
6.1.2	For PAE at the gaming machines, all necessary approvals for gaming machine variations must be obtained as required under section 3.5.5 of the Act. Note: *Submissions to this effect should be made in the manner required from time to time by the Commission*
6.1.3	A Tester must assess and provide a recommendation of the PAE to be installed on or in a gaming machine, including any cabling, brackets and fittings prior to any application for a variation to a gaming machine type to the Commission. Note: *A Tester is not required to test the installation of PAE on every machine, but they must test a sample of all PAE and its related components for every EGM type that it is intended for.*

The approval/authorisation process for PAE is detailed at *Appendix C – Approval Process for PAE.*

7 Document Information

7.1 Document details

Criteria	Details
Document title:	Victorian Player Account Equipment Technical Requirements Document
Document owner:	Victorian Commission for Gambling & Liquor Regulation
Document author:	Pre-commitment Implementation Project, OLGR, Department of Justice & Regulation

7.2 Version control

Version	Date	Description	Author
V1.0	December 2015	Public release	OLGR, Department of Justice
V1.1	June 2015	Public release	OLGR, Department of Justice & Regulation
V1.2	May 2016	Public release – for comment	OLGR, Department of Justice & Regulation
V2.0 draft	July 2016	Final for approval	OLGR, Department of Justice & Regulation
V2.0	December 2016	For publication post approval	OLGR, Department of Justice & Regulation / VCGLR

7.3 Approvals

Name	Position	Function
Commission	The Commission	Approve

8 Appendix A – Minimum specifications for PAE

Player Account Equipment		Technical Specifications	Power Supply	Security
Gaming Machine	**Card reader**	A reader capable of reading magnetic stripe cards with an encoded Track 2. Reader must be capable of reading magnetic stripe cards with low coercivity. Reader is to be capable of reading at least up to position 22 of Track 2. The reader must at least support commands for: • removal and insertion of a card • be able to read and output track data	SMIB (via USB only) or a power supply that is not sourced from monitoring equipment or gaming equipment power supply. *(N/A for Crown Casino)*	Access to internal components must be restricted and monitored.

	Interactive display	A touch screen graphic display capable of displaying colour images and textual content.
		
		Display may be:
		
		• side-mounted and facing the front of the EGM
		
		• mounted within the front of the EGM
		
		• picture-in-picture game screen overlay with touch screen capability.
		
		Display is to be capable of:
		
		• displaying:
		
		  • pre-commitment messages in a portrait orientation which:
		
		  ▪ must be at least 54mm wide and 144mm high and
		
		  ▪ must have a minimum resolution 150 pixels (wide) x 400 pixels (high)
			(N/A for Crown Casino)
			
			**Note: Power may be sourced from the monitoring SMIB power supply in a manner approved by the Monitoring Licensee*
			
			(Contact Monitoring Licensee for approved power sourcing methods)
				Access to internal components must be restricted and monitored.

Player Account Equipment		Technical Specifications	Power Supply	Security
		or • pre-commitment messages that are displayed in a landscape orientation which: ▪ must be a width of no less than 95mm or a height of no less than 54mm ▪ must have a resolution of 267 pixels (wide) x 150 pixels (high) • a cable(s) to facilitate connection to the Monitoring SMIB: • for touch interactivity, e.g. USB; and • for graphics display, e.g. VGA. All displays must present pre-commitment messages, menus, buttons and labels in a clearly visible and legible manner, without distortion and as presented by the PCS. All pre-commitment displayed messages, menus, buttons and labels must be clearly visible and legible by a person sitting at and standing in front of the gaming machine. Touch screens must be accurate so that user's actions are interpreted correctly.		
Kiosk	**General**	Computer device secured within a cabinet (kiosk) that may be: • Wall mounted; or, • Freestanding; or, • Desktop mounted	Venue power	The computer device component within the kiosk must be physically inaccessible to the general public. Administrative access is restricted to authorised

Player Account Equipment		Technical Specifications	Power Supply	Security
		A kiosk must be fitted with: • Keypad entry facility (physical or virtual) • Keyboard entry facility (physical or virtual) • Connection to the internet. Controlled website access using a restricted user interface or "shell" configuration.		support staff only. Pre-commitment data must pass through in a secure manner. Pre-commitment data must not be stored or modified for any purpose other than for the pre-commitment scheme.
	Display	The display must be of dimensions capable of displaying the pre-commitment website homepage in a navigable manner. The display resolution must be at a minimum of 1024 pixels x 768 pixels Touch screens must be accurate so that user's actions are interpreted correctly. Must clearly display and provide access to the pre-commitment system via a button on the "home screen" or "menu" page.	Kiosk or venue power	
	Card reader	Magnetic stripe card reader (swipe or insertion) Must be capable of reading a magnetic stripe cards with low coercivity Must be capable of reading at least to position 22 of Track 2 on a player card Must read cards that are consistent to ISO/IEC 7811 standards.	Kiosk	
	Style guide	Must comply with the Pre-Commitment Kiosk implementation style guide. Refer to Appendix E.	Not Applicable	
Card encoder/ reader and keypad	**Card encoder / reader**	A card encoding/reading device that is configurable to interface with the pre-commitment system website using either manual or automated data entry. Capable of writing to at least position 22 of Track	Venue	

Player Account Equipment		Technical Specifications	Power Supply	Security
		2 on a magnetic stripe card. (see Appendix B) Must be able to read and write to a magnetic stripe card with low coercivity. Must encode cards consistent to ISO/IEC 7811 standards. Must be consistent with the encoding convention detailed in Appendix B of this document.		
	Keypad	A keypad that is a physical or virtual device that enables the secure entry of a PIN	Venue	
Card	**Magnetic stripe card**	Magnetic stripe card able to be encoded consistent with the pre-commitment scheme personal identifier. The magnetic stripe needs to support at least low coercivity as specified in the ISO/IEC 7811 standard Must comply with the card size of ID-1 format as specified in ISO/IEC 7810 standard	N/A	

9 Appendix B – Player card configuration

Card Encoding Positions on Track 2

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22

Assignment of Track 2 Positions for Combined ID's and Pre-commitment Only ID

SS	Loyalty ID	Pre-commitment ID	ES	LRC

Loyalty and Pre-commitment ID's Combined with End Sentinel and LRC Following the Last Pre-commitment ID Character

;	0	1	2	3	4	5	6	7	8	9	10	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	?	LRC

Pre-commitment ID Only. All Loyalty Positions Must be Zero'd

;	0	0	0	0	0	0	0	0	0	0	0	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	?	LRC

Assignment of Track 2 Positions for Loyalty Only

SS	Loyalty ID	ES	LRC

Loyalty Only with End Sentinel and LRC Following the Last Loyalty ID Character

;	0	1	2	3	4	5	6	7	8	9	10	?	LRC

(Loyalty ID data is optional)

Note: Character in Position 1 must be set to "0" for all pubs and clubs and 1 or greater for Melbourne Casino loyalty ID's

Control Characters

SS **Start Sentinel** (Hex 0B) ;

LRC **Longitudinal Redundancy Check Character**

ES **End Sentinel** (Hex 0F) ?

Loyalty ID and Pre-commitment ID data must be written to positions 0 - 22 in ANSI/ISO BCD 5 bit format and comply with ISO 7811 standard.

10 Appendix C – Approval process for PAE

Player Account Equipment		Technical Requirements	Supplier	ATF recommendation	VCGLR approval	Installer
Gaming Machine	Card reader	Minimum specifications as specified in the Player Account Equipment Technical Requirements (PAETR)	Anyone	Requires ATF recommendation for approval of variation to approved gaming machines (tested for compliance with Regulations and Commission Standards). ATF is to certify that devices are consistent with PAETR.	Approval for variation to a gaming machine for each machine type is required. Approval number will be issued by VCGLR.	Installation must be performed by a technician who holds a gaming industry employees licence, or, holds the appropriate license and is employed by Crown Casino. Licensed technician to certify the equipment has been installed, interacts with the PCS, and meets its functional requirements.
	Interactive display					
Kiosk	Display	Minimum specifications as specified in the Player Account Equipment Technical Requirements (PAETR)	Anyone	Not required	VCGLR approval is not required Functional certification to be performed only.	Licensed technician to certify PAE interacts with the PCS and meets its functional requirements.
	Card reader					
Service Point Workstation	Card encoder/reader	Minimum specifications as specified in the Player Account Equipment Technical Requirements (PAETR)	Anyone			

Player Account Equipment		Technical Requirements	Supplier	ATF recommendation	VCGLR approval	Installer
	Keypad	Minimum specifications as specified in the Player Account Equipment Technical Requirements (PAETR)	Anyone			
Card	Magnetic stripe card	Minimum specifications as specified in the Player Account Equipment Technical Requirements (PAETR)	Anyone		N/A	N/A

11 Appendix D – PAE interface specification

As attached as version 1.5.

12 Appendix E – Kiosk implementation – Style guide

As attached as version 0.2.

Victorian Pre-Commitment System

PAE Interface Specification

Version 1.5

External

INTRALOT S.A.
Kifissias Ave. & 3 Premetis Str
151 25, Athens, Greece
Tel: +30 210 6156000,Fax:+30 210 6106800
www .intralot.com

INTRALOT GAMING SERVICES
299 Williamstown Road
Port Melbourne, Vic 3207, Australia
Tel +61 3 9673 3909, Fax:+61 3 9673 3999
Www.igsmonitor.com.au

(f) accepts no liability for any use of this document or reliance placed on it by any end user.

This document is provided on an 'as is' basis and, to the extent permitted by law, INTRALOT excludes all express and implied warranties or guarantees and all liability for any loss or damage, whether direct, indirect or consequential (including loss of profit, loss of opportunity, loss of contracts or loss of revenue) arising from:

(a) an end user's use of or reliance on this document;

(b) any inaccuracies contained in this document; and

(c) any non-compliance by any goods or services with any representation, illustration, specification or other information contained in this document,

whether such loss or damage arises from negligence or otherwise.

By relying on these specifications, end users release INTRALOT from all liability and accept sole responsibility associated with the use of the information and specifications in this document, irrespective of the purposes for which such use is applied.

Nothing in this disclaimer should be taken to exclude, restrict or modify the application of any condition, warranty, guarantee, right or remedy conferred or implied under the Competition and Consumer Act 2010 (Cth) including Schedule 2, the Australian Consumer Law or any other law, where to do so would contravene that law, or cause any part of this disclaimer to be void.

This document is current at the date of publication. INTRALOT reserves the right to withdraw, modify, update or replace this document at any time without notice.

SD Template v1.4

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Document Purpose

The Victorian Government is committed to implementing a voluntary pre-commitment scheme across Victoria commencing on December 1, 2015.

Intralot Gaming Services (IGS), the licensed Monitor, is undertaking to provide pre-commitment services for the Victorian Government.

This document details the technical specification for how Player Account Equipment (PAE) can interface with the IGS Pre-Commitment System (PCS) and also describes the process for suppliers to submit their PAE components for inclusion on the Compatible PAE List.

Appendix G contains details of a range of PAE components that have undergone preliminary testing for compatibility with the Pre-Commitment System.

As application is made for further components to be added to the Compatible PAE List, the details will be added to the list and amendments made to provide information as to the progress of testing. The Office of Liquor Gaming and Racing will publish the details of the Compatible PAE list.

1.2 Document Scope

This document covers the following:

- Technical specification for connecting the Intralot Monitoring/PCS SMIB to the Player Interface Module (PIM) components in EGMs not participating in a loyalty program;
- Technical specification for interfacing the Player Kiosk and Service Point PC card reader/encoder components with the PCS Web portal, for both loyalty and non-loyalty venues;
- Card reading and encoding functions;
- Process for suppliers to submit other components for inclusion on the Compatible PAE List;
- Details of the Monitoring/PCS SMIB USB port and power specifications for connecting the PAE components installed in EGMs.

1.3 Out of Scope

This document excludes the following:

- Technical specifications for connecting the Intralot Monitoring/PCS SMIB to PIM devices installed on an EGM that is participating in a loyalty program. Loyalty suppliers should contact IGS for the technical specification for the mandatory connection of the loyalty PIM equipment to the Intralot Monitoring/SMIB for pre-commitment purposes.

1.4 Glossary

Term or Abbreviation	Description
ATF	Accredited Testing Facility. A tester listed on the Roll of Manufacturers, Suppliers and Testers.
Certified Tester	A tester who has attained ISTQB Certified Tester Advanced Level (CTAL) accreditation and/or is a member of NATA (National Association of Testing Authorities, Australia), and/or is a tester on the Roll of Manufacturers, Suppliers and Testers.
Compatible PAE List	The list of PAE components that have been certified to meet the Victorian PAE Technical Requirements (R3) and that also meet PCS integration requirements. The list is maintained by IGS in conjunction with OLGR.
IE	Internet Explorer
OS	Operating System
PAE	Player Account Equipment
PCS	Pre-Commitment System
PIM	Player Interface Module
Player Service Point	A location in the venue where the PAE encoder/card reader and keypad is located.
Service Point PC	A PC that connects the card encoder/reader and keypad to the PCS Web portal application. It can be a new or an existing venue PC.
VCGLR	Victorian Commission for Gambling and Liquor Regulation.

YourPlay	This is the brand name chosen by the government for the Victorian Pre-Commitment System. References to "YourPlay" are interchangeable with "PCS".

Table 1: List of Glossary

1.5 Referenced Documents

References	Document Title	Version
R1	AUS/NZ Gaming Machine National Standards	V10.0
R2	Victorian Appendix to the AUS/NZ Gaming Machine National Standards	V10.0
R3	Victorian Player Account Equipment Technical Requirements (December 2014)[1]	
R4	Kiosk Implementation Style Guide	V1.0

Table 2: Referenced Documents

[1]Available from the VCGLR website
http://www.vcglr.vic.gov.au/home/laws+and+regulations/policy/technical+standards/

2 EGM PAE Components

2.1 Overview

This section describes the technical specification for connecting the Intralot Monitoring/PCS SMIB to the Player Interface Module (PIM) components in an EGM not participating in a loyalty program.

2.2 Player Interface Module (PIM)

The Player Interface Module (PIM) components that must be installed and have satisfied a pre-commissioning functional test on each EGM include the following components:

- Magnetic card reader,
- Display monitor,
- Touchscreen functioning in association with the display monitor,
- Mounting accessories and cabling harness.

2.3 Connections

The PIM components are connected directly to the Intralot Monitoring/PCS SMIB. The interconnection is shown in the following diagram:

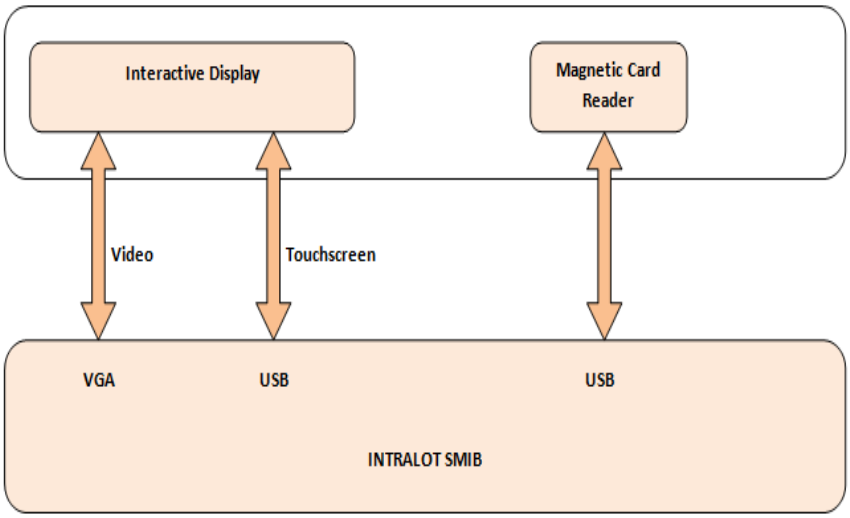


Figure 1: Interconnection between Monitoring SMIB and PIM components

2.4 SMIB External Components Connection

The connection types are listed in the table below:

External Components	Interface
Interactive Display Video Screen	VGA port The VGA cable to be used for the connection of the PAE touch monitor onto the monitoring SMIB must have the following pins disconnected as these signals are not supported by the monitoring SMIB: Pin 12 – DDC2B SDA Pin 15 – I2C SCL
Touch Screen	USB port
Magnetic Card Reader	USB port

Table 3 : SMIB External Components Connection

2.5 SMIB Power Specifications

The details of the SMIB power specifications are provided in Appendix F.

2.6 Digital Display Panels

The display component of the PIM is an integrated module that is connected via a standard VGA port to the Monitoring/PCS SMIB.

The integrated VGA controller provides the interfacing to the digital LCD panel as well as the image adjustments for brightness and contrast. In addition it provides an interface between the analogue signal timing and the digital panel timing.

The PCS menu screens are designed to be displayed on colour LCD panels that are 6.2” (640 x 240 pixel resolution) in a landscape orientation. Details of the digital display panel components that have undergone preliminary testing for compatibility with the PCS are shown in Appendix G.

Other components will be added provided that they comply with the following requirements and then follow the process described in Section 5.

2.6.1 Kernel Timings

The current SMIB OS Kernel caters for two display timings. Components with alternative display timings may require changes to the SMIB OS Kernel.

See Figure 2 below for the current timing resolutions:

LINUX name	Display option 1	Display option 2
Refresh	60	60
Xres	640	640
Hres	240	240
Pixclock	40000	44543
Left_margin	138	300
Right_margin	22	300
Upper_margin	208	22
Lower_margin	67	25
hsync_len	6	50
vsync_len	2	3
Sync	FB_SYNC_HOR_HIGH_ACT |FB_SYNC_VERT_HIGH_ACT	FB_SYNC_CLK_LAT_FALL
Vmode	FB_VMODE_NONINTERLACED	FB_VMODE_NONINTERLACED

Figure 2 : Kernel Timings for the Integrated Panels

2.6.2 Resolution

Only Digital Display Panel models of 640 x 240 pixel resolution (landscape) that meet the display panel requirements in the PAE Technical Requirements (R3) will be considered for PCS integration as described below:

2.6.2.1 Same Timing/Resolution

Digital Display Panel models of 640 x 240 pixel resolution (landscape) that follow the same timing signal/resolution as those shown in Figure 2 will be able to be integrated with PCS subject to successful verification and interoperability testing as described in Section 5.

2.6.2.2 Different Timing/Resolution (SMIB Kernel Change Required)

Digital Display Panel models of 640 x 240 pixel resolution (landscape) with different timing signal/resolutions to those shown in Figure 2 will only be considered on the basis of their inclusion in a future upgrade of the kernel as described in Section 5.

This will need discussion with IGS as any SMIB kernel change will require an extensive cycle of testing to verify that the monitoring application has not been affected and there may be a significant delay before the next scheduled kernel upgrade.

2.7 Touchscreens

Touchscreens for the interactive display can be of any of the known technologies, such as resistive, capacitive, SAW (surface acoustic wave), provided that they can be interfaced using a touchscreen controller compatible with the Linux operating system used by the Monitoring SMIB.

2.8 Touchscreen Controllers

Details of the touchscreen controller components that have undergone preliminary testing for compatibility with the PCS are shown in Appendix G.

Other touchscreen controller components will be able to be integrated provided that they comply with the following requirements and then follow the process described in Section 5.

2.8.1 Touchscreen Controller Requirements

- Compliant with the Victorian Pre-commitment Player Account Equipment Technical Requirements (R3);
- Able to work on Linux Kernel version 2.6.35.3;
- Must be compliant with the Linux input subsystem and its event interface library (TSLIB) and export via USB any touch to the input module in the form of events and pass it via TSLIB (raw data) to the application;
- Compliant with non X environment;
- As PCS requires high availability, the touchscreen driver must be robustly

connected to the system and be able to re-connect quickly and transparently in case of disconnections. For this reason, the touch controller driver must be a Kernel driver with driver source code available for IGS compilation and customization purposes;

- The compiled driver must be able to be downloaded from the CMCS Host remotely;
- Connection via standard USB interface to the Monitoring/PCS SMIB must be compliant with USB version 2.0 certified Type A connection;
- If powered via USB it should not exceed the maximum power supply of 500mA via the SMIB USB port. If the component requires a supply above 500 mA, an alternative power supply must be sourced;
- Proposed component(s)/model(s) should not have any end of life issues and will continue to be supported by the manufacturer.

2.8.2 No SMIB Kernel Change

Touchscreen controllers that do not require any changes to the SMIB OS Kernel will be able to be integrated, subject to successful verification and interoperability testing.

2.8.3 SMIB Kernel Change Required

- Touchscreen controllers that require a change to the SMIB Linux OS kernel will only be considered on the basis of their inclusion in a future upgrade of the kernel;
- This will need discussion with IGS as any SMIB kernel change will require an extensive cycle of tests to verify that the monitoring application has not been affected and there may be a significant delay before the next scheduled kernel upgrade.

2.9 Magnetic Card Readers

Details of the magnetic card reader components that have undergone preliminary testing for compatibility with the PCS are shown in Appendix G.

Other magnetic card reader components will be able to be integrated provided that they comply with the following requirements and then follow the process described in Section 5.

2.9.1 Magnetic Card Reader Requirements

- The Magnetic Card Reader must comply with the Victorian Pre-commitment Player Account Equipment Technical requirements (R3);
- Must work on Linux OS Kernel version 2.6.35.3;
- Must have a USB-to-Serial (USB CDC) type of connection. USB HID devices can also be integrated but will need to be examined per case;
- The communication interface protocol must be in compliance with both the Linux OS and the PIM Card Reader API described in Appendix B;
- Electrical connection should be via a USB port compliant with USB version 2.0 certified Type A connection;
- If powered via USB it should not exceed the maximum power supply of 500mA via the SMIB USB port. If the component requires a supply above 500 mA, an alternative power supply must be sourced;
- Proposed component(s)/model(s) should not have any end of life issues and will continue to be supported by the manufacturer.

2.9.2 No SMIB Software Change

Magnetic Card Readers that do not require any changes to the SMIB software will be able to be integrated with PCS, subject to successful verification and interoperability testing.

2.9.3 SMIB Software Change Required

- If a driver and/or a library for implementing an API are required, the source code must be made available to IGS;
- The compiled driver/library must be able to be downloaded from the CMCS Host remotely;
- The component will only be considered on the basis of inclusion in a future upgrade of the SMIB software.

3 Player Service Point PAE Components

3.1 Overview

The Player Service Point PAE comprises a Keypad and Card Reader/Encoder connected to a Service Point PC.

The Service Point PC allows the card reader/encoder to connect to the Pre-Commitment System via the PCS Web portal application. The Card Reader/Encoder may be a single component or a separate card reader and card encoder.

The Service Point PC must meet the following minimum requirements:

- Operating System: Windows XP (SP 3), Windows 7 and Windows 8;
- Browser: Google Chrome and IE
    - For Windows 7 and Windows 8, the minimum supported browser is IE11 or Google Chrome. The Google Chrome browser is recommended for optimal results;
    - For Windows XP the supported browser is Google Chrome.

  Intralot does not guarantee results with other OS/browser combinations.

3.2 Venue Portal Security

The communication between the Service Point PC and the PCS Venue Portal (http://yourplay.igsmonitor.com.au/vp) uses the standard HTTP protocol and secured using VPN. The Service Point PC Web API commands use the HTTP protocol.

3.3 Player Service Point Keypad/ Pinpad

The Player Service Point must have a Keypad/ Pinpad connected to the Service Point PC. This is used to allow players to key in their PCS PINs.

The keypad must be a numeric keypad with a USB connection that complies with USB keyboard input devices standard.

3.4 Card Reader/Encoder

The Service Point PC must have card reader and card encoder capabilities. This may be provided by either separate card reader and card encoder components or a combined card reader/card encoder component.

The card reader/encoder is used by venue staff to read or encode the PCS ID card for the player.

The card can be either a dedicated PCS ID card or a dual card for PCS and loyalty.

- The card encoder may be a combined reader/encoder component or a printer/encoder component that can print on the card at the same time that it encodes the magnetic stripe. The card encoder is not required to print on the card;
- The card encoder allows the encoding of the player's card according to the standards set by the PAE Technical Requirements (R3);
- The card encoder obtains the player's PCS ID from the PCS Web Portal application for encoding on the PCS card;
- The card reader obtains the player's PCS ID from the card and passes it to the PCS Web portal application.

Details of the Service Point PC Card Readers/Encoders that have undergone preliminary testing for compatibility with the PCS are shown in Appendix G. Other card reader/encoder components may be able to be integrated provided that they are verified through the process described in Section 6.

3.5 Non-Loyalty Venue Card Encoding Function

For non-loyalty venues, the Venue operator will be able to encode a PCS only card through the PCS Web Portal encoding function using the integrated card encoder devices.

The non-loyalty encoder application will not cater for capturing a Loyalty ID or encoding a dual card. Only the PCS ID is encoded on the PCS card and the loyalty ID is padded with zeros.

3.6 Loyalty Venue Card Encoder Application

The Loyalty supplier will need to upgrade the loyalty card encoder application to perform the following functions on the Service Point PC:

- Encode a new card with a PCS ID only i.e. without a loyalty ID;
- Encode a new dual access card (PCS and Loyalty);
- Encode a PCS ID on an existing loyalty card;
- Encode a new loyalty card without a PCS ID[2];
- Replace a lost card;
- Replace/Refresh a damaged card.

The venue staff can use copy/paste or manual entry methods to acquire the PCS ID from the PCS Web portal (Strongly not recommended by the State due to potential errors).

[2] The loyalty card encoder application must follow the ANSI/ISO 7811 standard to append the ES (End Sentinel) and the LRC (Longitudinal redundancy check) after the 11 digit Loyalty ID (for a Loyalty only card), and after the 9 digit PCS ID for other cards

4 Venue Kiosk

4.1 Overview

The Kiosk is a self-service terminal with Internet access and is located inside each venue. The Kiosk allows players to access the PCS Web portal and to login to their account using their PCS Card/PIN or their username/password.

The Kiosk terminals used for PCS may be:

- Pre-commitment only Kiosks or
- Combined Loyalty and Pre-commitment Kiosks.

A Kiosk printer is optional and PCS does not support printing at the venue Kiosk.

Supported Kiosk OS/Browser combinations:

- Operating system: Windows XP[3] (SP3), Windows 7, Windows 8;
- Browsers: Google Chrome and IE
    - For Windows 7 and Windows 8, the minimum supported browser is IE11 or Google Chrome;
    - The Google Chrome browser is recommended for optimal results across all Windows operating systems.

Intralot does not guarantee results with other OS/browser combinations.[4]

4.2 Kiosk Browser Security

National and Victorian security standards dictate that the communication of authentication details between the Kiosk browser and the PCS portal must be secure. As a result, the Kiosk PCS Web Site must use HTTPS for authentication details. IGS has developed a solution that allows the Kiosks Web API commands to be sent in

[3] XP with IE8 is **strongly** not recommended by Intralot.

[4] There may be compatibility problems with other operating system and/or web browser combinations that are not specified here. As such Intralot cannot guarantee the correct operation of the pre-commitment Kiosk website on non-specified operating system and/or web browser combinations.

HTTP or HTTPS mode, depending on the browser used. See Section 10.3 for more information.

4.3 Kiosk Card Reader

The Kiosk Card reader is a magstripe card reader able to read Track 2 of any ANSI/ISO7810/11 card. The card reader can be either a swipe, half or full insert type provided that it can read from character 1 up to and including 20 of Track2.

The 20 characters do not include the control characters listed below which are in addition to the 20 characters:

- Start Sentinel, (SS)
- End Sentinel, (ES)
- LRC

Details of the Kiosk card readers that have undergone preliminary testing for compatibility with the PCS are shown in Appendix G. Other Kiosk card reader components may be able to be integrated provided that they are verified through the process described in Section 6.

4.4 Kiosk Reader Application

Kiosk suppliers can interface the Kiosk with the PCS Web Portal by either implementing the Intralot specified WebAPI (Appendix D) or the Magnetic Card API (Appendix E) in the Kiosk shell to allow the player's card to be read at the Kiosk. Refer to Appendix C for guidance on selecting the appropriate API.

The WebAPI allows the PCS and/or loyalty application to poll for the card reader status and card ID for the last card inserted/swiped. This will support asynchronous queries to the application coming from both loyalty and PCS applications in response to the card insertion/swiping.

5 Process for Including PIM Components on the Compatible PAE List

The process for suppliers to nominate PIM components that meet the specifications in Section 2 for inclusion on the Compatible PAE List is described below:

5.1 All PIM Components

- The component must comply with the Victorian Pre-commitment Player Account Equipment Technical Requirements (R3) in all respects and must be consistent with this interface specification;
- The supplier must complete the details in the "Magnetic Card Reader and Interface Display Specification Form" included in Appendix A and submit the form to IGS;
- IGS will perform an initial investigation to ascertain whether the component can be integrated without any SMIB kernel changes.

5.2 No SMIB Kernel/Software Changes Required

1. The supplier must provide IGS with the component hardware and driver or driver source code for testing.
2. IGS will discuss timeframes for interoperability testing and any costs with the supplier and schedule the testing.
3. IGS will perform interoperability testing of the component.
4. If the testing is satisfactory:
    - IGS will notify the supplier of the testing result;
    - IGS will add the component to the Compatible PAE List;
5. If the testing is unsatisfactory:
    - IGS will provide the supplier with details of the testing results and the supplier may decide to apply changes and resubmit for testing.

5.3 SMIB Kernel/Software Changes Required

If changes to the SMIB OS Linux kernel/software are required:

1. The supplier will be notified.
2. IGS will provide the supplier with details of the SMIB kernel/software upgrade timetable.
3. IGS will discuss timeframes for interoperability testing and any costs with the supplier.
4. IGS will recompile the SMIB kernel/software with the source code provided by the supplier in accordance with the planned SMIB kernel/software compilation schedule.
5. IGS will perform interoperability testing of the component.
6. If the testing is satisfactory:
   - IGS will notify the supplier of the testing result;
   - IGS will add the component to the Compatible PAE List;
   - Intralot will arrange ATF testing of the SMIB baseline changes to meet the planned SMIB kernel/software upgrade schedule.
7. If the testing is unsatisfactory:
   - IGS will provide the supplier with details of the testing results and the supplier may decide to make changes and resubmit for testing.

6 Process for including non-Loyalty PAE Components on the Compatible PAE List

The process for nominating a Service Point PC card reader/encoder component or Kiosk card reader for inclusion on the Compatible PAE List is described below:

1. The component must comply with the Victorian Pre-commitment Player Account Equipment Technical Requirements (R3) in all respects and must be consistent with this interface specification.
2. The supplier must implement one of the following methods to interface with the Web portal application:
   - The Intralot specified WebAPI in Appendix D: or
   - The Intralot specified PCS Magnetic Card API in Appendix E to produce a DLL driver to interface with the Web Portal at the ExeApp level.

   Refer to Appendix C for guidance on API use.
3. The supplier must engage a Certified Tester to perform interoperability testing of the component. (Intralot will provide a link to a PCS testing environment to facilitate this testing at a date to be confirmed).
4. If testing is satisfactory, the certified tester will notify the supplier and IGS.
5. The component will be included on the Compatible PAE List.

7 End to End Workflows and Message Flow Diagrams

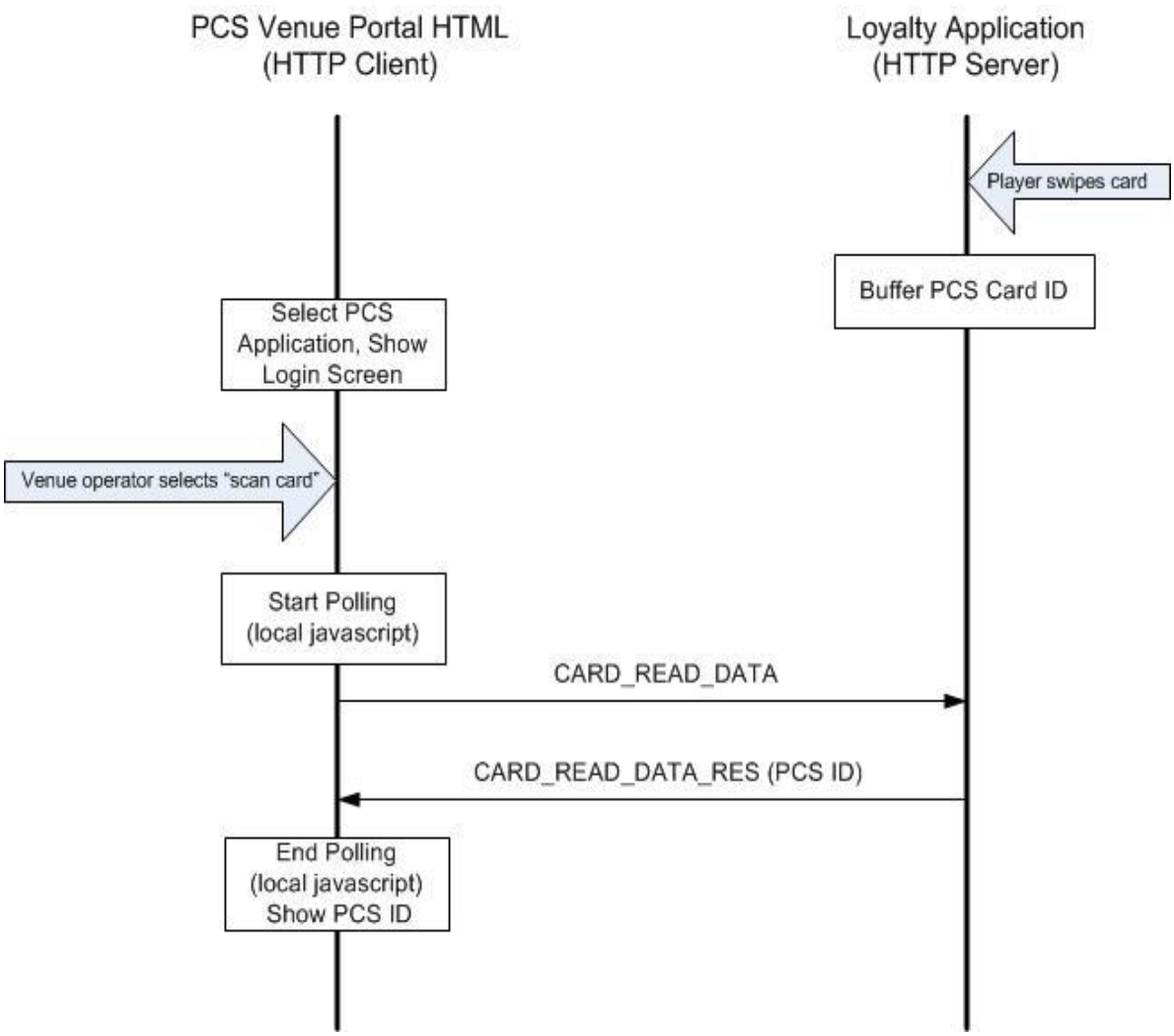
7.1 Player Service Point PC

7.1.1 Workflow Diagram – Encode a Card

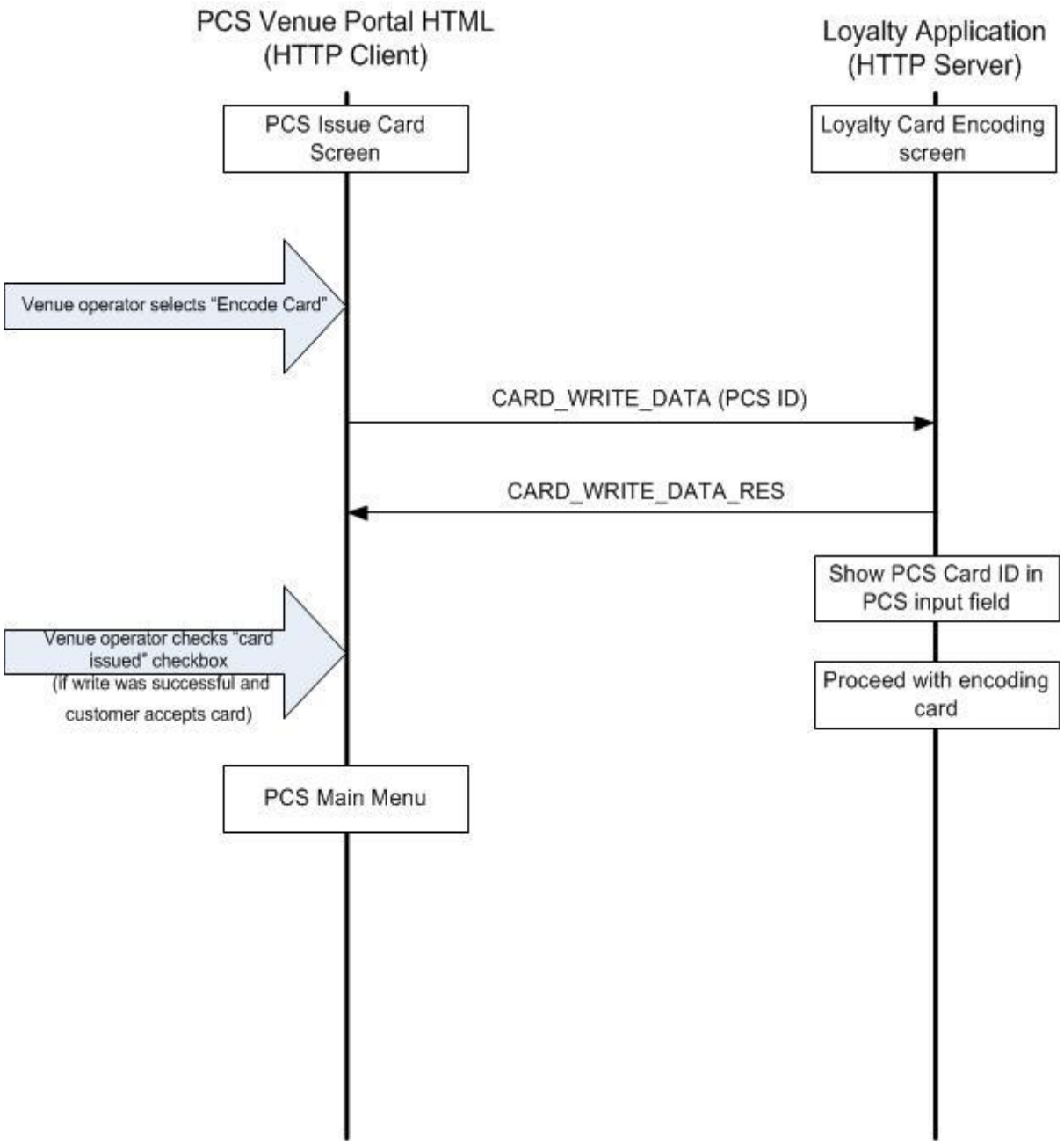
The workflow below covers the following scenario:

- To encode a card.

7.1.2 Message Flow Diagram - Venue PC (Read Function)



7.1.3 Message Flow Diagram - Venue PC (Encode Function)



7.1.4 Responsibilities of a Loyalty Provider (Service Point PC)

The Loyalty Provider must:

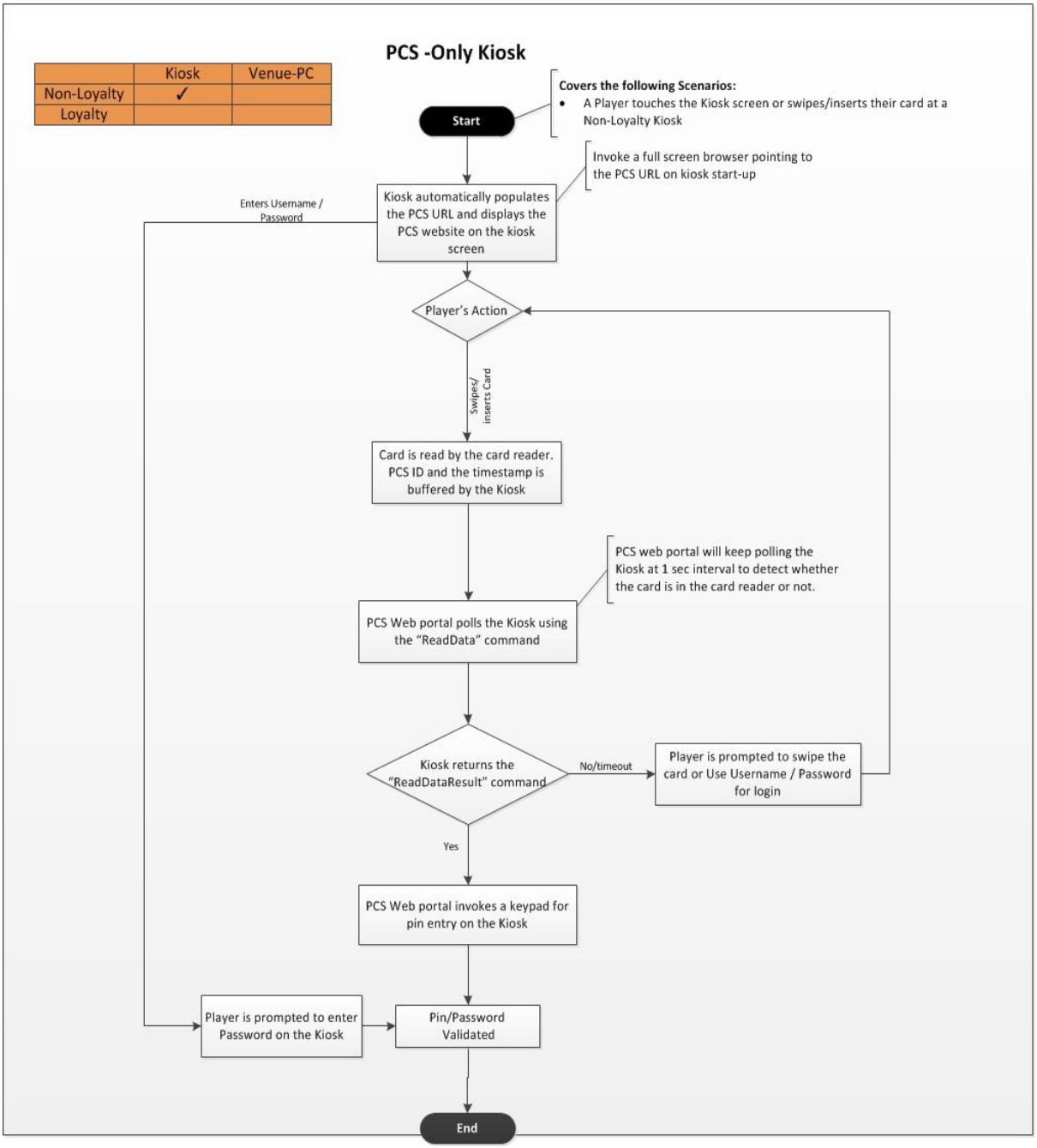
- Follow the work flow diagrams and the message flow diagrams specified in Section 7.1.1 – 7.1.3;
- Provide the Loyalty application on the Service Point PC which must be able to perform the following encoding functions on the same PC:
  - Encode a Dual access card i.e. Loyalty + PCS;
  - Encode a Pre-commitment only card;
  - Encode a Loyalty only card;
  - Replace a lost card;
  - Replace/Refresh a damaged card

7.2 End to End Work Flows for the Kiosk

7.2.1 Workflow Diagram– PCS Only Kiosk

The flow below covers the following scenario:

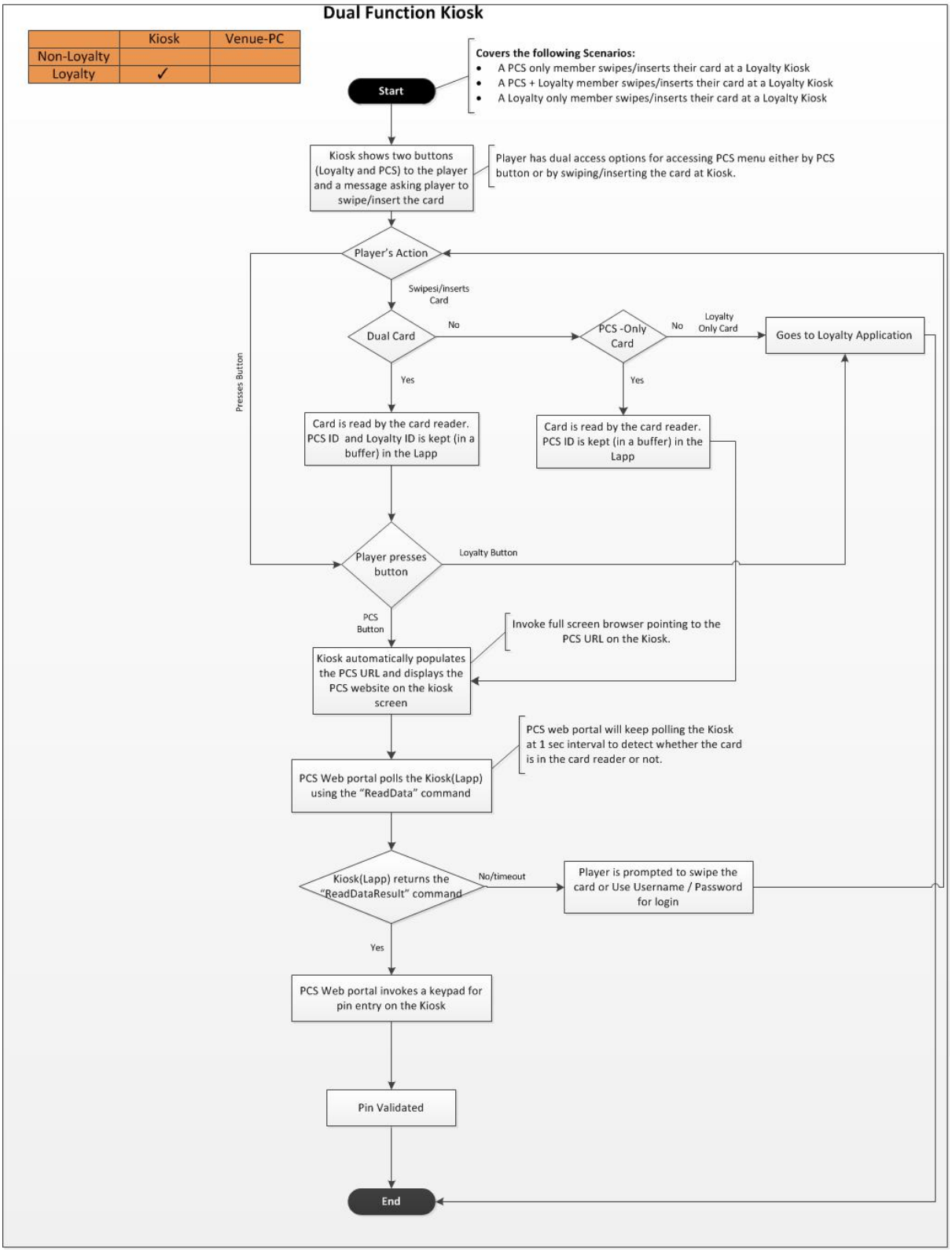
- A member swipes/inserts his card at a Non-Loyalty Kiosk.



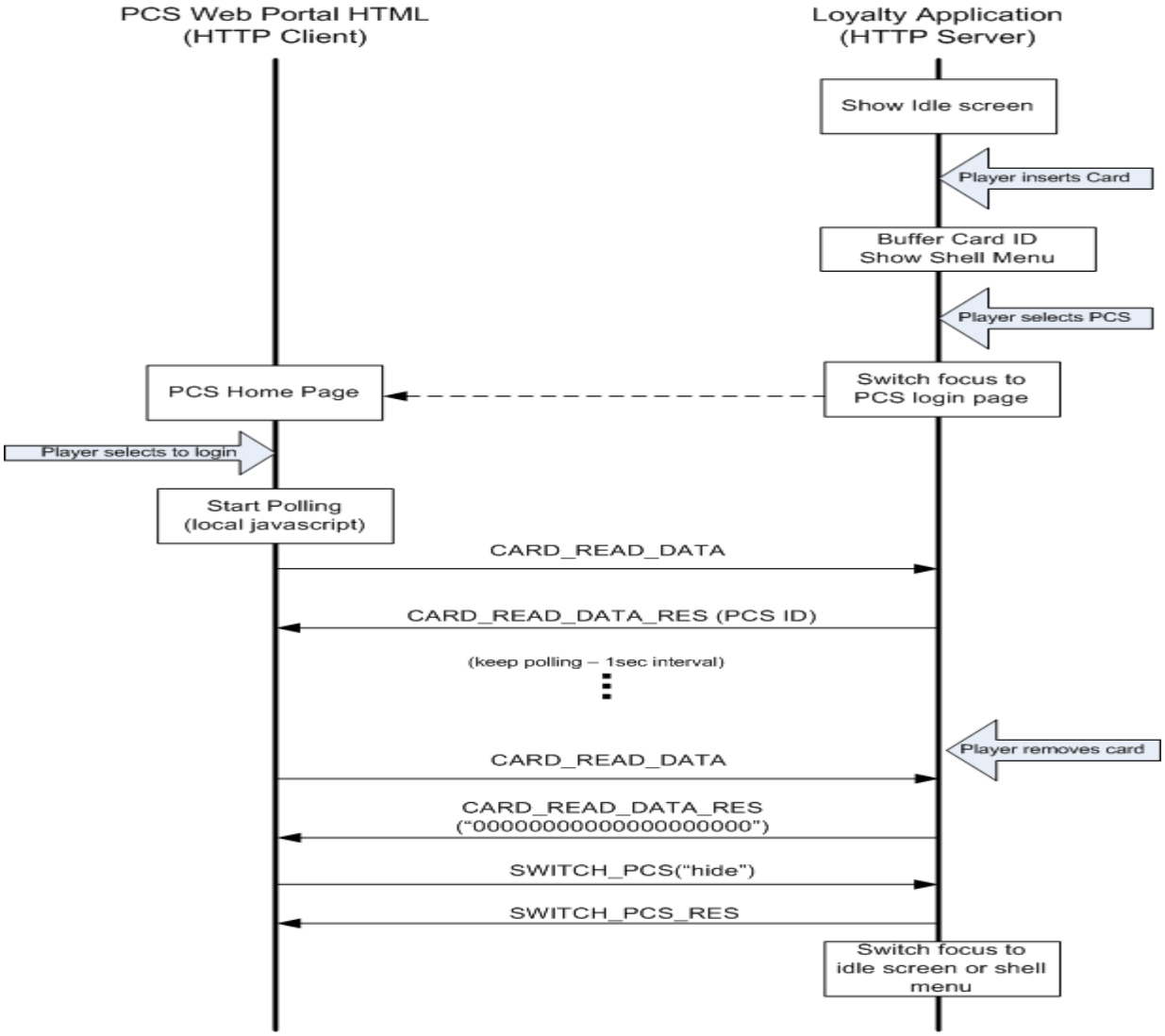
7.2.2 Workflow Diagram– Dual Function (Loyalty) Kiosk

The workflow below covers the following scenario:

- A member swipes/inserts his card at a Loyalty Kiosk;



7.2.3 Message Flow Diagram - Venue Kiosk Message Exchange (Login and Logout – Insert Card)[5]



In reply to every CARD_READ_DATA poll, CARD_READ_DATA_RES returns:

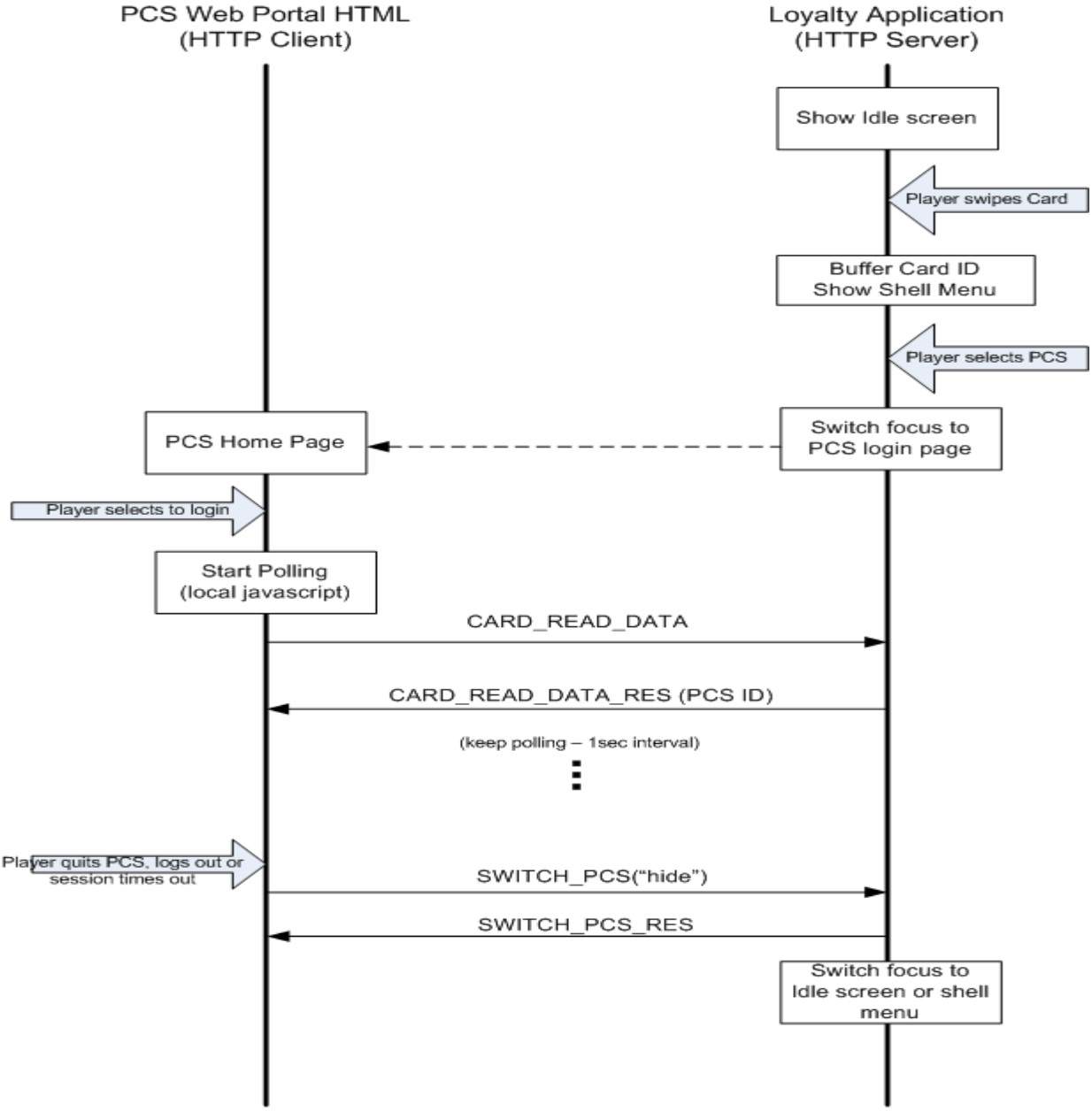
- card inside the reader: the PCS ID (field: "data"), and the time of the last insert (field "time").
- no card in reader: a string with 20 zeroes (field: "data"), and the time of the last removal (field "time").

See also section 11.2

- If there has not been any previous card activity, "data" and "time" can be empty strings ("").

[5] The references to HTTP refer to both HTTP and HTTPS.

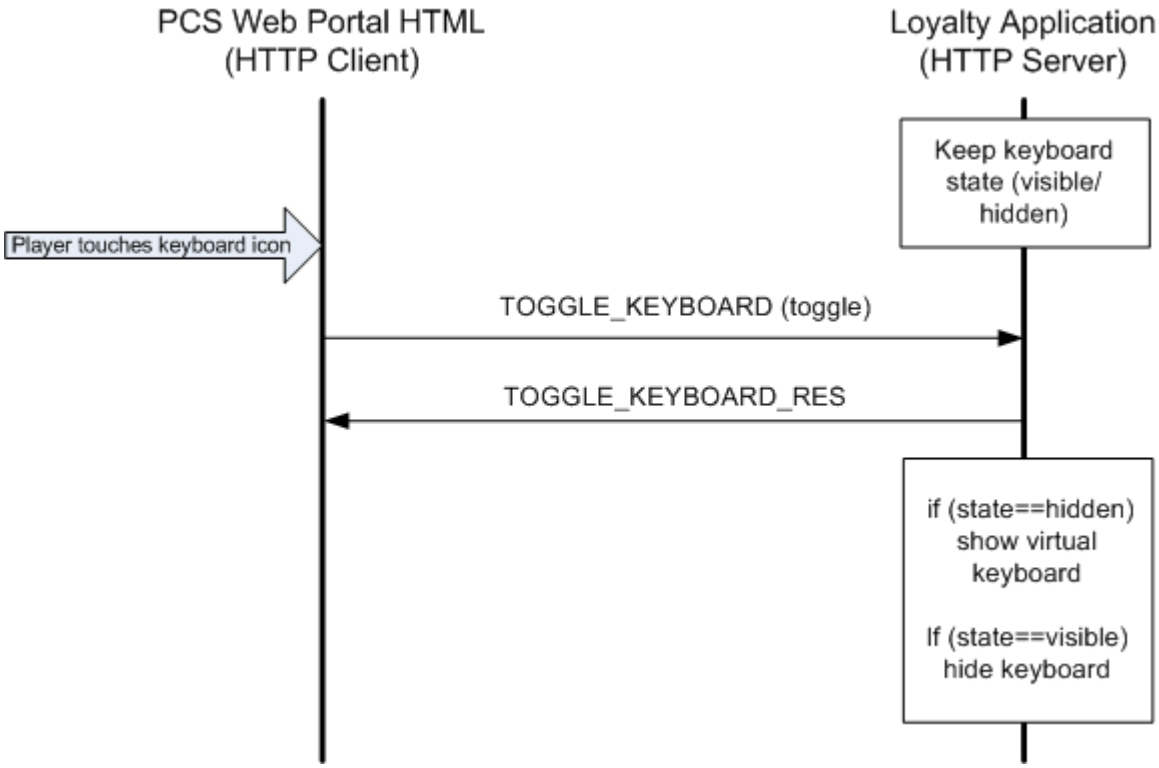
7.2.4 Message Flow Diagram - Venue Kiosk Message Exchange (Login and Logout – Swipe Card)[6]



- In reply to every CARD_READ_DATA poll, CARD_READ_DATA_RES returns the PCS ID of the last card that was swiped through the reader (field: "data") and the time of the last swipe (field "time"). See also section 11.2.
- If there has not been any previous card activity, "data" and "time" can be empty strings ("").

[6] The reference to HTTP refers to both HTTP and HTTPS.

7.2.5 Message Flow Diagram - Venue Kiosk Message Exchange (Toggle Keyboard)[7]



[7]. The references to http refer to both http and https

7.2.6 Responsibilities of the Kiosk Provider

The Kiosk Script must follow the work flow diagrams and the message flow diagrams specified in Section 7.2.1 – 7.2.5 of this document. The Kiosk provider should also meet the following requirements:

- Ensure that the technician installs a predefined ini file containing the venue id when he installs the Kiosk in the venue;
- Invoke a full screen browser pointing to the PCS URL on kiosk start-up. Use of iFrame is not recommended;
- Parse the predefined ini file containing the Venue ID and pass it as a parameter in the PCS URL, eg (https://yourplay.com.au/web/kiosk/home?venueid=<ID_OF_VENUE>&dual=true&sec=true).
    1. The PCS URL security parameter (sec) will differ depending on the browser that is being used. (See Sections 10.3 and 10.4). Ensure that the requirements regarding security and the use of HTTP/HTTPS described in Section 4.2 and Chapter 10 are followed;
    2. Implement the Pre-commitment touch screen keyboard as defined in the "Kiosk Implementation Style Guide (R4)";
- Show/Hide the onscreen keyboard via the "Toggle Keyboard" command when requested by the PCS Web Portal. The Kiosk application must be able to show the keyboard upon receiving the next "Toggle Keyboard (show/hide)" command;
    1. Disable all OS keyboard shortcut combinations (e.g. Ctrl-Alt-Delete) on the keyboard in order to avoid any player interaction with the OS;
    2. Disable the browser option to save passwords.
- For card insert reader kiosks, if a card is not present in the reader, the Kiosk application must return a string of 20 zeroes ("00000000000000000000") as the PCS ID argument when polled by the web portal "Read Card" command;
- Ensure that the Kiosk returns to the Idle screen when one of the following occurs:

1. The player logs out of PCS;
2. The player session expires (i.e. No Activity Timeout);
3. The player removes the card from the card reader or
4. No Touch Timeout is enabled

- The use of "no touch timeout" is recommended but is not mandatory. If it is implemented, the Kiosk application must log out the player if the player has logged in;
- Player must be able to swipe/insert the card on the Kiosk main menu and the PCS Kiosk portal. The Kiosk application must ensure that it can detect a card swipe/insert and respond to the CARD_READ_DATA poll with the PCS ID of the card when the PCS Kiosk portal is displayed(see also the diagram in section 7.2.4);
- The kiosk display of "Pre-commitment Services" must not be less prominent than other ancillary services also displayed;
- In case of communication failure, show a meaningful message ("PCS Web site currently unavailable") and offer the ability to attempt to reconnect ("Try again" button).

8 Appendix A - Magnetic Card Reader and Interactive Display Specifications Form

Form instruction

Please refer to the Example/Explanation column for information on how to fill in the form.

1. Please cut and paste the component specification table if you intend to support multiple Components of the same type.
2. This form does not cover picture in picture technology.
   The form is for informational purposes only and there is no commitment from the Pre-commitment service provider that it will be able to interface with Components specified in this form or with the ones referenced as example.
3. When you have completed this form, send it with any attachments to PAE@igsmonitor.com.au
4. For further information regarding this form, please contact Wilson Lo on 03-96733985.

Component Supplier Details (Must be provided)				
1.	Supplier Name:			
2.	Supplier Contact Name:	Phone:	Mobile:	Email :
3.	Number of components included in this form			
	Display Monitor:	Touch Screen Controller:	Magnetic Card Reader:	

8.1 Display Monitor Specifications

#	Description	Specification	Example / Explanation
1.	Connector Type (s)		VGA (mandatory)
2.	Screen resolution @ refresh rate (list all that apply)		- 640x240 @60Hz (native resolution) - 640x480 @60Hz
3.	Viewing area size		149 x 54 mm, 6.2” diagonal
4.	LCD panel manufacturer & model		HITACHI TX16D11VM2CQC
5.	Video timing parameters		Provide LCD panel datasheet which

			must include parameter values below: - Panel resolution (pixels) - Pixel clock (Mhz) - Horizontal Front porch (pixels) - Horizontal back porch (pixels) - Vertical Front porch (pixels) - Vertical back porch (pixels) - Hsync length (pixels) - Vsync length (pixels)
6.	Power and fuse input requirements		12VDC, 1A, Fuse rating 1.5A
7.	Grounding specification		Connected to chassis ground or specify otherwise
8.	Remarks (Optional)		

8.2 Touchscreen Controller Specifications

#	Description	Specification	Example / Explanations
1.	Manufacturer		3M
2.	Connector Type		USB (mandatory)
3.	Touch controller driver		- Must support Embedded Linux (Kernel version 2.6.35.3 or higher) - Be compliant with the Linux input subsystem and its event interface (TSLIB). - Kernel driver - Non X environment
4.	Power input requirements		- 5VDC, 100mA - USB powered within USB power limits (max 500mA) or state otherwise
5.	Remarks (Optional)		

8.3 Magnetic Card Reader Specifications

#	Description	Specification	Example / Explanations
1.	Manufacturer		PANASONIC
2.	Model Number		ZU-M2242S3R2
3.	Connector Type		USB (mandatory)
4.	Power input requirements		- 5VDC, 0.1A - USB powered within USB power limit (max 500mA) or state otherwise
5.	Driver type		USB-to-Serial (mandatory)
6.	Driver OS		Must support Embedded Linux OS (Kernel version 2.6.35.3 or higher)
7.	API specification (Refer to Appendix B)		API library must support the functions as described in Appendix B. **Note:** Certain functions are optional depending on the component. For this reason, they have a return error code meaning "Not supported".
8.	Remarks (Optional)		

9 Appendix B – PIM Card Reader API

NR	FUNCTION	DESCRIPTION	INPUT PARAMETERS	OUTPUT PARAMETERS
1	intOpenDevice (char * port)	Opens the port of the device (i.e. /dev/ttyACM0) and initializes the device	The string of the node	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of reader's reply failed 50: Command Timeout
2	intCloseDevice()	Closes the port		0: Success 10: Port cannot be closed
3	intGetDeviceVersion(char* constFWVersion)	Gets from the device the F/W version	FWVersion: buffer which is filled with device's F/W version	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of reader's reply failed 50: Command Timeout
4	intSendCardDirection(int direction)	It sets the direction of the card's read	direction: one byte which can be one of the following 1: insertion 2: withdrawal 3: both directions	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout
5	intGetReaderStatus(int *const status)	Get the device status	status: returns the status of the reader Following statuses should be supported: 1: Track 1 decode status (0 no data, 1 track 1 data exist) 2: Track 2 decode status (0 no data, 1 track 2 data exist) 3: Track 3 decode status (0 no data, 1 track 3 data exist) 4: Card present (0 no card present, 1 Card present)	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout
6	intReaderReset()	Resets the device to initial state, clears all buffers and re-initializes the device	9.1	0: Success 10: Port not opened 20: Device not found 30: Write of command failed

				40: Read of command failed 50: Command Timeout
7	intReadCardID(intconst* CardDataLength, char const* CardID)	It reads and returns the Card data from the track which has been set in SetTrackSelection function	CardDataLength: the length of the CardID string CardID: string in which card data are filled. The card data should be read from the track which is defined by SetTrackSelection function. If more than one track is defined the card data should be separated with the character defined in SetTrackSeparator function 9.2	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout
8	intSetDeviceReading(int reading)	It enables or disables the device's reading	reading: can be one of the following: 1: enable reading 0: disable reading	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout
9	intSetTrackSelection(int track) **(See Note)**	Sets the track from which the reader will read cards	track: bit-oriented hexadecimal number of the track to be set i.e. 30: any track 31: track 1 only 32: track 2 only 33: track 1 and track 2 34: track 3 only 35: track 1 and track 3 36: track2 and track 3 37: all 3 tracks	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout 60: Function Not supported
10	intSetTrackSeparator(int separator) **(See Note)**	It sets the character to be used to separate data decoded by a Multi-track reader.	separator: any ASCII character which is used to separate the data from different tracks	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout 60: Function Not supported

11	intLedControl(int colour) **(See Note)**	Sets the device led to specific colour	Colour: integer which defines the colour 1: green 2: red 3: orange	0: Success 10: Port not opened 20: Device not found 30: Write of command failed 40: Read of command failed 50: Command Timeout 60: Function Not supported

Note: Functions 9, 10 and 11 are optional, depending on the device. For this reason, they have an extra return error code (-60)

10 Appendix C – Service Point PC and Kiosk APIs

10.1 Overview

Intralot provides two APIs to allow the Service Point PC PAE and Kiosk PAE to interface with the Pre-commitment Web Portal Application (PCSApp).

PAE suppliers will need to implement one of the following APIs on the Service Point PC and Kiosk depending on their configuration:

- A PCS Magnetic Card API (Appendix E) describing how to interface with the PCS Web Portal at the ExeApp level. ExeApp is a download executable file supplied by Intralot that interfaces the PCSApp with the Intralot integrated Card Encoder/Card reader components or
- A WebAPI describing how to interface with the Pre-commitment Web portal application (Appendix D). This consists of a simple and minimal set of JSON commands that are communicated using the standard HTTP protocol.

10.2 API Selection

The following two configurations allow PAE suppliers to integrate their magnetic card components with PCS:

(a) by using the Intralot supplied ExeApp and

(b) without using the ExeApp application.

10.2.1 With ExeApp

This configuration is applicable to PAE suppliers who choose to use the Intralot provided ExeApp application to provide an end-to-end interface between the PCSApp and their unintegrated card reader/encoder component using one of the following methods:

- The PAE supplier will need to provide a DLL driver by implementing Intralot's Magnetic Card API (Appendix E). The Magnetic Card API is used for interfacing the ExeApp with the card reader and card encoder components. Each component has a DLL that drives it and exports the Magnetic Card API

functions through which the ExeApp reads, writes, gets status, etc. for the card component;

- Alternatively, the PAE supplier can implement the WebAPI "client part" (Appendix D) to interface with the ExeApp to access the integrated card reader/encoder. In this case the ExeApp is the HTTP parser that handles the WebAPI messages originating from PAE supplier application (e.g. Loyalty Application "Lapp" in Figure 3).

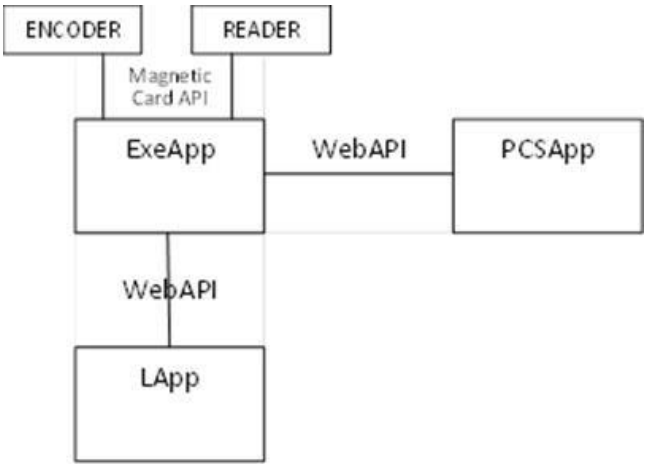


Figure 3 : PCS Web Portal connection to the Intralot integrated Encoder/Reader components via the ExeApp application

10.2.2 Without ExeApp

This configuration is applicable to PAE suppliers who choose to support their components in their own system, without PCS directly interfacing with the card reader/encoder component and without using the Intralot provided ExeApp application.

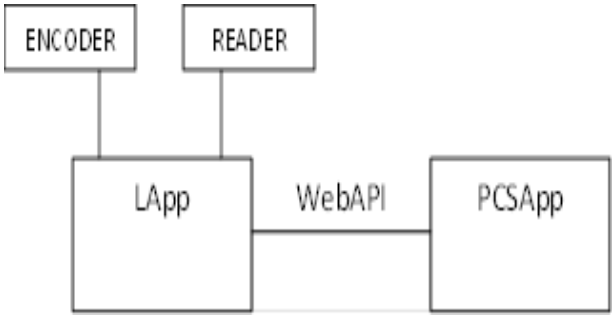


Figure 4 : PCS Web Portal connection to the Loyalty application that drives the Encoder/Reader

- The WebAPI consists of a simple and minimal set of JSON commands that are communicated using the standard HTTP protocol;
- The PAE supplier's application will have to provide the HTTP parser (like a minimal Web server) and implement the WebAPI "server part" in order for the PCSApp Web client to be able to access the magnetic card component;
- The PCSApp will initiate a HTTP Post transaction to the PAE supplier's application (e.g. Loyalty Application "Lapp" in Figure 4), which drives the card reader and card encoder. E.g. PCSApp sends the HTTP Post ReadData to Lapp , it polls the hardware and returns the ReadDataResult.

10.3 Kiosk Configuration

The following parameters need to be specified at the Kiosk in order to be passed via the URL to the PCS Web Portal (eg (https://yourplay.com.au/web/kiosk/home?venueid=<ID_OF_VENUE>&dual=true&sec=true):

- venueid: The ID (numeric) of the venue that the Kiosk is located. Will be provided by Intralot. If one venue has multiple kiosks, all kiosks will have the same venue ID.
- dual: Can have values "true" or "false". Denotes whether this kiosk also runs a loyalty and/or other applications requiring a shell menu (value="true"), or only runs a browser pointing permanently to PCS (value="false").
- sec: This should have the value "true" to indicate that the Web API is running in secure mode (HTTPS). If the value is "false" this indicated that the Web API is running in HTTP mode. If this parameter is not used, it defaults to "false".

The above parameters should be stored in a location which allows them to be edited by the technician who sets up the Kiosk for the venue, e.g. C:\KioskConf.ini.

10.4 Kiosk Configuration Security Settings

The communication of authentication details between the Kiosk browser and the PCS Web Portal must be secured (HTTPS). IGS has developed a solution that allows the Kiosks Web API commands to be sent in HTTP or HTTPS mode, depending on the browser used.

10.4.1 Google Chrome Browser

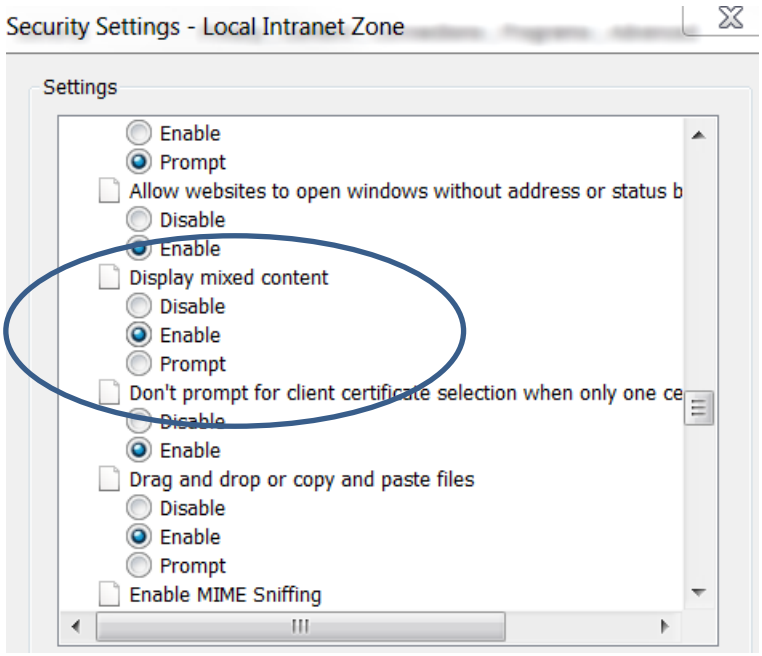
As Google Chrome supports mixed HTTPS/HTTP mode, the following options are available for Kiosks using Google Chrome:

Option 1 (Mixed HTTPS/HTTP mode)	Option 2 (Secure HTTPS mode)
Use Kiosk Web API HTTP commands using Chrome running in mixed mode with the flag "--allow-running-insecure-content" in chrome.exe	Change the Web API to support HTTPS and add the [sec=true] parameter to indicate that the Web API is running in secure mode

IGS recommends Kiosk providers to build support for the Web API using HTTPS mode, as this will address the security concern of supporting mixed content within the Web browser.

10.4.2 IE Browser

- For Kiosks using the IE browser, the implementation must accept HTTPS Kiosk Web API commands over SSL as IE browser does not support mixed HTTPS/HTTP mode;
- To do this, the parameter [sec=true] must be added to the PCS URL, eg https://yourplay.com.au/web/kiosk/home?venueid=<ID_OF_VENUE>&dual=true&sec=true;
    - If the parameter is [sec= false] or if the [sec] parameter is not used, the Web API commands will continue to be sent using HTTP;
- The browser security setting for "Display mixed content" must also set to "Enable"


Security Settings - Local Intranet Zone
Settings
Enable
Prompt
Allow websites to open windows without address or status b
Disable
Enable
Display mixed content
Disable
Enable
Prompt
Don't prompt for client certificate selection when only one ce
Disable
Enable
Drag and drop or copy and paste files
Disable
Enable
Prompt
Enable MIME Sniffing

11 Appendix D– WebAPI[8]

11.1 Background

11.1.1 PCS_EXE Protocol

All http messages consist of a start-line followed by a sequence of one or more header lines and a message body.

HTTP-message = start-line

*(header-line CRLF)

CRLF

[message-body]

The normal procedure for parsing an HTTP message is to read the start-line into a structure, read each header field into a hash table by field name until the empty line, and then use the parsed data to determine if a message body is expected.

An "IGS HTTP message" is like an ordinary "HTTP message" with specific body. In order for a server to be able to distinguish a PCS_EXE HTTP request among other HTTP requests, all PCS_EXE HTTP messages will use one of the following start-lines:

11.1.2 POST /PCS_EXE HTTP/1.1

The "Content-Type" in the headers should be: "Content-Type: pcs_exe_api_v1_0/json" in order to be parsed as an CS_EXE API HTTP message. The v1_0 signifies the version of the API and is changed accordingly.

Also the header-line characterized by the "Content-Length" field will specify the size of the PCS_EXE HTTP body which follows and will be described shortly (json message). The server response for such a request should be to write on the same connected socket a PCS_EXE HTTP body with the corresponding response.

Finally the "Access-Control-Allow-Origin" header value should be set to "*".

[8] The references to HTTP refer to both HTTP and HTTPS

The Kiosk and Service Point PC should run a local HTTP service listening for PCS Web API request on port 8020. The browser pointing to the PCS Web Application will run a local javascript which talks to this service by sending JSON messages to 127.0.0.1:8020.

11.1.3 PCS_EXE HTTP BODY FORMAT

A PCS_EXE HTTP BODY message has the Json format (http://www.json.org/).
The general syntax is as follows:

```
"{
        "msgheader":
        {
                "CommandID":"COMMAND_ID_VALUE",
                "TrnsTime": seconds_since_1_1_1970 // int time of transaction
        },
        "params":
        {
        "ParamName1": "ParamValue1",
        "ParamName2": "ParamValue2",
        …
        }
}"
```

11.2 API Messages

The syntax of the API messages is as follows:

11.2.1 ReadData:

```
"{
        "msgheader":
        {
                "CommandID":"CARD_READ_DATA",
                 "TrnsTime": seconds_since_1_1_1970 // int time of transaction
        }
}"
```

11.2.2 ReadDataResult:

```
"{
        "msgheader":
        {
                "CommandID":"CARD_READ_DATA_RES",
                "TrnsTime": seconds_since_1_1_1970 // time of transaction
        },
        "params":
        {
                "time": "seconds_since_1_1_1970",  // time card was read
                "result": the_result  //int (e.g. 0 for Success - rest can be found in document
        R1)
                "data": "the_track_data"   //string – return track data from position 1
        }
}"
```

11.2.3 WriteData:

```
"{
        "msgheader":
        {
                "CommandID":"CARD_WRITE_DATA",
                "TrnsTime": seconds_since_1_1_1970 // int time of transaction
        },
        "params":
        {
                "length": the_length_of_data,  // int
                "data": "the_track_data"   //string – encode track data from position 1
        }
}"
```

11.2.4 WriteDataResult:

```
"{
	"msgheader":
	{
		"CommandID":"CARD_WRITE_DATA_RES",
		 "TrnsTime": seconds_since_1_1_1970 // int time of transaction
	},
	"params":
	{
		"result": the_result  //int (e.g. 0 for Success - rest can be found in document
		R1)
	}
}"
```

11.2.5 Status:

```
"{
	"msgheader":
	{
		"CommandID":"CARD_STATUS",
		"TrnsTime": seconds_since_1_1_1970 // int time of transaction
	}
}"
```

11.2.6 StatusResult:

```
"{
        "msgheader":
        {
                "CommandID":"CARD_STATUS_RES",
                 "TrnsTime": seconds_since_1_1_1970  // int time of transaction
        },
        "params":
        {
                "encoderStatus": "idle" OR "write_pending" OR "write_success" OR
        "write_failed",
                "encoderVersion": "132",
                "encoderManu": "Manufacturer",
                "encoderModel": "ModelType",
                "readerStatus": "idle" OR "read_pending" OR "read_success" OR "read_failed",
                "readerVersion": "847",
                "readerManu": "Manufacturer",
                "readerModel": "ModelType"
        }
}"
```

11.2.7 ToggleKeyboard:

```
"{
        "msgheader":
        {
                "CommandID":"TOGGLE_KEYBOARD",
                "TrnsTime": seconds_since_1_1_1970 // int time of transaction
        },
        "params":
        {
                "toggle": "show", "hide", "toggle"  //only "show" and "hide" will be used

        }
}"
```

11.2.8 ToggleKeyboardResult:

```
"{
        "msgheader":
        {
                "CommandID": "TOGGLE_KEYBOARD_RES",
                "TrnsTime": seconds_since_1_1_1970 // time of transaction
        },
        "params":
        {
        "result": the_result   // int (0: success, other: fail)
        }
}"
```

11.2.9 Switch PCS Screen:

```
"{
        "msgheader":
        {
                "CommandID":"SWITCH_PCS",
                 "TrnsTime": seconds_since_1_1_1970 // int time of transaction
        }
        "params"
        {
                "visibility": "show" or "hide"
        }
}"
```

11.2.10 Switch PCS Result:

```
"{
        "msgheader":
        {
                "CommandID":"SWITCH_PCS_RES",
                 "TrnsTime": seconds_since_1_1_1970 // int time of transaction
        }
        "params"
        {
                "result": the_result                    //int (0: for Success, other: fail)
```

```
        }
}"
```

12 Appendix E– Service Point PC and Kiosk Magnetic Card API

The below API functions are provided for the card reader and card encoder manufacturers. For each device a corresponding DLL file should be implemented that will export all the functions mentioned in the following API. For the encoding devices, there is no need to provide both reading and encoding DLLs, since the reading functionality is incorporated in the encoding DLL.

Every function mentioned in the API uses blocking execution, thus the manufacturers should create DLLs that will provide thread safety.

The DLLs should be named according to the naming scheme "MyCompany_MyModel.dll". They should have versioning information added, which can be extracted through the use of Windows API functions. If the manufacturer's DLLs require external libraries or DLLs, that don't need installation, then these should be located "side-by-side" with those DLLs. If the manufacturer's DLLs depends on external files that needs installation, then the manufacturer must also provide these installers and all the necessary files. In that case the manufacturer also must provide documentation, describing all the necessary steps for the installation procedure.

The location of the DLL files in order for the Venue PC Application to locate them is: C:/Ireni/Devices/

12.1 CARD READER API FUNCTIONS

Syntax	***int OpenReader(char * port)***
Parameters	**port**: String of characters of the device's port name. Example: In the case of a RS-232 device, the port name should be "COMxx" (e.g. "COM3", "COM12"). If the device has a USB connection then the port name should be "USB". In this particular case, this function is responsible to discover the device (e.g. by enumerating the connected USB devices).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Opens the port of the reader and initializes the device.

Prototype	***int CloseReader()***
Parameters	None
Return value	0: Success -10: Port not opened -20: Device not found
Description	Closes the port and releases all resources.

Syntax	***int GetReaderManufacturer(char * const manufacturer)***
Parameters	**manufacturer**: Buffer which is filled with device's manufacturer name (maximum size is 256 characters).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device the manufacturer's name Note:The function should be available when the module starts (before opening the port).

Syntax	***int GetReaderModel(char * const model)***
Parameters	**model**: Buffer which is filled with device's model type (maximum size is 256 characters).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device the model type Note:The function should be available when the module starts (before opening the port).

Syntax	***int GetReaderType(int * type)***
Parameters	**type**: 1: swipe 2: insertion
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device the reader type.

Syntax	***int GetReaderVersion(char * const fw_version)***
Parameters	**fw_version**: Buffer which is filled with device's firmware version (maximum size is 256 characters).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device its firmware version.

Syntax	***int SetReaderCardDir(int direction)***
Parameters	**direction**: 1: insertion 2: withdrawal 3: both directions
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Configures the reading direction of the card.

Syntax	***int GetReaderStatus(int * const status)***
Parameters	**status**: Returns the status of the reader. The following status flags should be supported: Bit 1: Track 1 decode status (0 no data, 1 track 1 data exist) Bit 2: Track 2 decode status (0 no data, 1 track 2 data exist) Bit 3: Track 3 decode status (0 no data, 1 track 3 data exist) Bit 4: Card present (0 no card present, 1 Card present)
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Gets the device's status.

Syntax	***int ReaderReset()***
Parameters	None
Return value	0: Success -10: Port not opened

	-20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Resets the device to initial state, clears all buffers and re-initializes the device.

Syntax	***int ReaderEnable(int state)***
Parameters	**state**: 1: enable reading 0: disable reading
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Enables or disables the device's reading ability.

Syntax	***int SetReadTrackSelection(int track)***
Parameters	**track**: Bit-oriented number of the track(s) to be set i.e. 0x30 (hex): any track 0x31 (hex): track 1 only 0x32 (hex): track 2 only 0x33 (hex): track 1 and track 2 0x34 (hex): track 3 only 0x35 (hex): track 1 and track 3 0x36 (hex): track2 and track 3 0x37 (hex): all 3 tracks
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Selects the tracks that the reader will read from the card.

Syntax	***int SetReadTrackSeparator(int separator)***
Parameters	**separator**: Any ASCII character which is used to separate the data from different tracks.
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout

Description	Sets the character to be used at the end of each track, to facilitate the separation of the data received from the reader.

Syntax	***int ReadTrackData(int const * track_length, char const * track_data, time_t * timestamp)***
Parameters	**timestamp**: The exact time when the track_data was read. **track_length**: The length of the track_data string. **track_data**: String in which card data are filled. The card data should contain only the tracks selected by *SetReadTrackSelection* function. For every track selected the track data should have the separator character appended, which is defined with the *SetReadTrackSeparator* function. If a track selected has no data then only the track separator should exists. Example: Track 1 and 2 are selected with track separator '\n'. The card read finds data only in track 1. The read operation should return "12345\n\n".
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	It reads and returns the card's data from selected tracks.

Syntax	***int ReaderLedControl(int colour)***
Parameters	**colour**: 1: green 2: red 3: orange
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout -60:Function Not supported
Description	Sets the device led to specific color. Note: The function is optional, depending on the device. For this reason, it has an extra return error code (-60)

12.2 CARD ENCODER API FUNCTIONS

Syntax	***int OpenEncoder (char * port)***
Parameters	**port**: String of characters of the device's port name. Example: In the case of a RS-232 device, the port name should be "COMxx" (e.g. "COM3", "COM12"). If the device has a USB connection then the port name should be "USB". In this particular case, this function is responsible to discover the device (e.g. by enumerating the connected USB devices).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of encoder's reply failed -50: Command Timeout
Description	Opens the port of the reader and initializes the device.

Syntax	***int CloseEncoder()***
Parameters	None
Return value	0: Success -10: Port cannot be closed
Description	Closes the port and releases all resources.

Syntax	***int GetEncoderManufacturer(char * const manufacturer)***
Parameters	**manufacturer**: Buffer which is filled with device's manufacturer name (maximum size is 256 characters).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device the manufacturer's name. Note: The function should be available when the module starts (before opening the port).

Syntax	***int GetEncoderModel(char * const model)***
Parameters	**model**: Buffer which is filled with device's model type (maximum size is 256 characters).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed

	-40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device its model type Note: The function should be available when the module starts (before opening the port).

Syntax	***int GetEncoderType(int * type)***
Parameters	**type**: 1: swipe 2: insertion
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device the encoder type.

Syntax	***int GetEncoderVersion(char * const fw_version)***
Parameters	**fw_version**: Buffer which is filled with device's firmware version (maximum size is 256 characters).
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of reader's reply failed -50: Command Timeout
Description	Gets from the device its firmware version.

Syntax	***int SetEncoderCardDir(int direction)***
Parameters	**direction**: 1: insertion 2: withdrawal 3: both directions
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Configures the encoding and/or reading direction of the card.

Syntax	***int GetEncoderStatus(int * const status)***
Parameters	**status**: Returns the status of the reader. The following status flags should be supported: Bit 1: Track 1 decode status (0 no data, 1 track 1 data exist) Bit 2: Track 2 decode status (0 no data, 1 track 2 data exist) Bit 3: Track 3 decode status (0 no data, 1 track 3 data exist) Bit 4: Card present (0 no card present, 1 Card present)
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Get the device's status.

Syntax	***int EncoderReset()***
Parameters	None
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Resets the device to initial state, clears all buffers and re-initializes the device.

Syntax	***int EncoderEnable(int state)***
Parameters	**state**: 1: enable device 0: disable device
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Enables or disables the device's writing and reading ability.

Syntax	***int SetTrackSelection(int track)***
Parameters	**track**: bit-oriented number of the track(s) to be set i.e. 0x30 (hex): any track 0x31 (hex): track 1 only 0x32 (hex): track 2 only 0x33 (hex): track 1 and track 2 0x34 (hex): track 3 only

	0x35 (hex): track 1 and track 3 0x36 (hex): track2 and track 3 0x37 (hex): all 3 tracks
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Sets the card's tracks that the reader will read or write to.

Syntax	***int EraseCard()***
Parameters	None
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Will erase all the tracks of the card.

Syntax	***int SetTrackSeparator(int separator)***
Parameters	**separator**: any ASCII character which is used to separate the data from different tracks.
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Sets the character to be used at the end of each track, to facilitate the separation of the data received from or encoded by the device.

Syntax	***int WriteTrackData(int const * track_length, char const * track_data)***
Parameters	**track_length**: the length of the track_data string **track_data**: string in which card data are filled. The card data will be written to the tracks which are defined by *SetTrackSelection* function. For every track selected the track data should have the separator character appended, as defined with the *SetTrackSeparator* function. If a track selected has no data then only the track separator should exists.
Return value	0: Success

	-10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Encodes the selected track data to the card.

Syntax	***int ReadTrackData(int const * track_length, char const * track_data, time_t * timestamp)***
Parameters	**timestamp**: the exact time when the track_data was read. **track_length**: the length of the track_data string. **track_data**: string in which card data are filled. The card data should be read from the tracks which are defined by *SetTrackSelection* function. For every track selected the track data should have the separator character appended, as defined with the *SetTrackSeparator* function. If a track selected has no data then only the track separator should exists. Example: Track 1 and 2 are selected with track separator '\n'. The card read finds data only in track 1. The read operation should return "12345\n\n".
Return value	0: Success -10: Port not opened -20: Device not found -30: Write of command failed -40: Read of command failed -50: Command Timeout
Description	Reads and returns the card's data from the selected tracks.

13 Appendix F – SMIB Power Specifications

The SMIB power supply is rated at 12V, 3.0A and is used for powering up the SMIB itself as well as the peripherals that can be connected on its external ports. The SMIB power supply unit must only be used to power integrated PAE components.

A summary of the powering scheme for the PAE equipment is shown in the table below:

Type	Connection	Power model
Display	VGA	External power or powered from the SMIB power supply following the SMIB Power Splitter requirement below.
Touchscreen and Touch controller	USB	External power or powered from the USB following the USB Interface power requirement below.
Magnetic Card reader	USB	External power or powered from the USB following the USB Interface power requirement below.

13.1 USB Interface Power Requirement

- Any component connecting via standard USB interface must be compliant with the USB version 2.0 certified Type A connection;
- If powered via USB, the component should not exceed the maximum power consumption standard for the USB of 500mA per port.

13.2 SMIB Power Splitter Requirement

- The maximum power assigned to the PAE Display Panel is 12W maximum, i.e. 12V/1A;
- The PAE supplier must provide a power splitter cable for ATF approval that follows the drawing diagram in Appendix H. The detailed cable specification is as follows:
    - The cable must have wires of at least 18AWG;
    - The power splitter cable must be equipped with an inline fuse (slow blow) rated at 1.0A towards the display or the monitor must have a respective fuse of 1.0A.

- In any case the splitter cable must establish that the output voltage to the Monitoring SMIB side and consequently input voltage to the SMIB will not drop below 11V @2A current draw. For that purpose the length/thickness of the splitter cable must be such so that above minimum voltage levels are always met.

14 Appendix G – Interim Component List

Details of the PAE components that have undergone preliminary testing for compatibility with the Pre Commitment System are shown below:

14.1 PIM Display Panels

Preliminary testing has indicated that the following Digital Display Panels will interoperate directly with the PCS/Monitoring SMIB, i.e. no further driver development work is needed to integrate with PCS, given that a VGA connection is used following the Kernel timing Option 1 or 2 as specified in Section 2.6.1:

Digital Panel Brand/model	Resolution/timing	Video standard
Kyocera/ TCG062HVLDA-G20[9]	640*240/60Hz	HVGA
Hitachi/ TX16D20VM5BQA[10]	640*240/60Hz	HVGA

14.2 PIM Touchscreen Controllers

Preliminary testing has indicated that the following Touchscreen Controllers will interoperate directly with the PCS/Monitoring SMIB:

Brand/model	Connection type	Power
3M SC401U[11]	USB HID	USB self-powered
DMC DUS1000 [12]	USB HID	USB self-powered

The connection type and Kernel driver information regarding the 3M touchscreen controller is provided below as an example:

Type	Brand/model	Connection	Driver
4-wireresistive touchscreen controller	3M Touch SC401U	USB HID	Kernel

[9] http://www.kyocera.co.uk/index/products/lcds_glass_glass_touch_panels/download.-cps-7018-files-56236-File.cpsdownload.tmp/TCG062HVLDA-G20Eng..pdf

[10] http://www.avnet-embedded.eu/fileadmin/user_upload/Files/Displays/Colour_TFT/TX16D20VM5BQA.pdf

[11] http://solutions.3m.com/wps/portal/3M/en_US/Electronics_NA/Electronics/Products/Touch_Systems/~/3M-MicroTouch-Electronics-EX-4-wire-Resistive-Controller-USB?N=5153292+3294736499&rt=rud

[12] http://www.datadisplay-group.de/fileadmin/pdf/produkte/Touchcontroller/DMC/DUS1000_Datasheet.pdf

14.3 PIM Magnetic Card Readers

Preliminary testing has indicated that the following Magnetic Card Readers will interoperate directly with the Intralot PCS/Monitoring SMIB:

Brand/model	Connection type	Power
MAGTEK Half Card Reader[13]	USB HID	USB self-powered
IDTECH Gaming reader[14]	USB HID	USB self-powered

14.4 Service Point PC Keypad/Pinpad

Any numeric keypad with a USB connection that complies with USB keyboard input devices standards may be used.

The following keypad is given as an example:

Brand/Model	Connection	Device type
KONIG Numeric keypad	USB	Keyboard

14.5 Service Point PC Card Printer/Encoders

Preliminary testing has indicated that the following Service Point PC Card Printer/Encoders will interoperate directly with the PCS Web Portal application, i.e. no further API development work is needed to integrate with PCS:

Brand/Model	Type	Encoding capacity
Datacard SP25[15]	Printer/Encoder	Track 2, Full capacity
IDTech EzWriter/EconoWriter[16]	Encoder/Reader	Track 1,2 ,3 Full capacity

[13] http://www.magtek.com/v2/products/secure-card-reader-authenticators/half-card.asp

[14] http://www.idtechproducts.com/products/insert-readers/117.html

[15] http://www.datacard.com/id-card-printers/sp25-plus-id-card-printer

[16] http://www.idtechproducts.com/products/swipe-readerwriters.html

14.6 Service Point PC Card Readers

Preliminary testing has indicated that the following Service Point PC Card Reader models will interoperate directly with the PCS Web Portal application:

Brand/Model	Type	Connection
IDTech EzWriter/EconoWriter[17]	Encoder/Reader	USB
IDTech Minimag Duo[18]	Reader	USB

14.7 Venue Kiosk Card Readers

Preliminary testing has indicated that the following two Kiosk Card Reader models will interoperate directly with the PCS Web Portal:

Brand/model	Connection type	Power
MAGTEK Half Card Reader[19]	USB HID	USB self-powered
IDTECH Gaming reader[20]	USB HID	USB self-powered

[17] Same link as Footnote 10 above.

[18] Same link as Footnote 10 above.

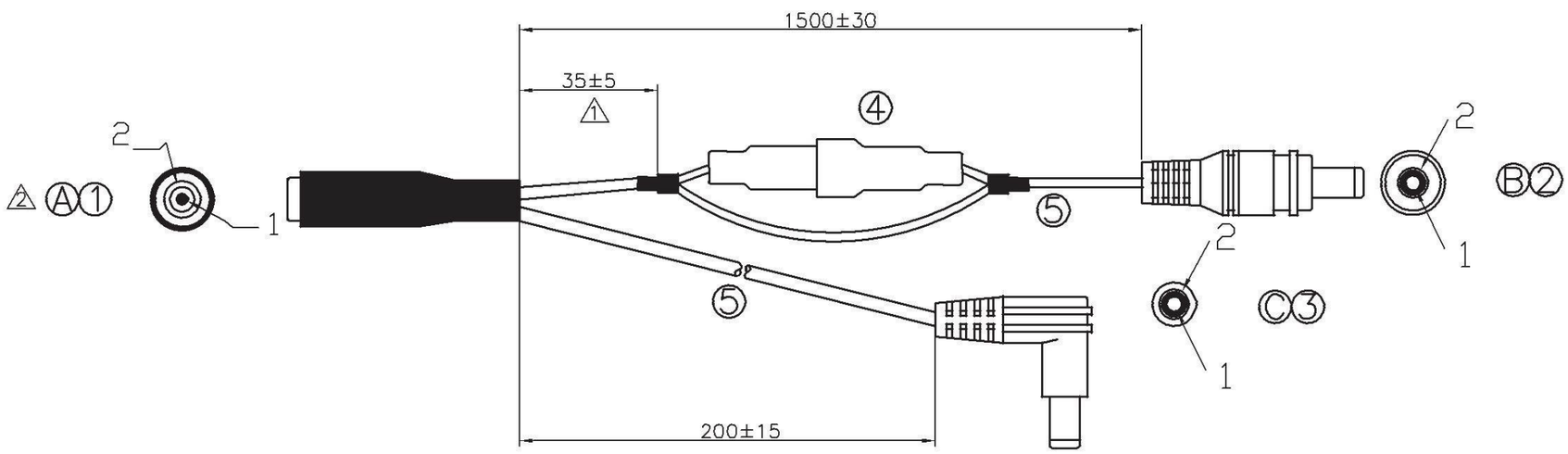
[19] http://www.magtek.com/v2/products/secure-card-reader-authenticators/half-card.asp

[20] http://www.idtechproducts.com/products/insert-readers/117.html

15 Appendix H – Power Splitter Cable Diagram

Customer Approved

ROHS



PIN ASSIGNMENT			
A	COLOR	B	C
1	CONDUCTOR	1	1
2	SPIRAL SHIELDING / FUSE	2	2

ITEM	MATERIAL
5	UL1185 18AWG*1C+S OD:ø3.5 JACKET:PVC BLACK
4	FUSE: 0229001P+01500274Z + HEAT SHRINK TUBE: ø4*30 BLACK*2PCS
3	DC PLUG ø5.5*ø2.5 L:23.5 MOLDING: PVC 45P BLACK 90° (MOLD NO. KH10202)
2	DC PLUG ø5.5*ø2.5 L:23.5 MOLDING: PVC 45P BLACK (MOLD NO. KH100246)
△2 1	DC JACK ø5.5*ø2.5+ HEAT SHRINK TUBE: ø8*40 BLACK

Rev.	Date	Description
△2	SEP/16/2014	CHANGE DC JACK ø5.5*ø2.1 → ø5.5*ø2.5
△1	SEP/01/2014	ADD DIMENSION

Approved		Part no	EU0111-103073101
Checked		Item	AS28000250
Drawing	Hunter	Drawing no	PDED_EU0111-103073101
Customer	EU0111	Unit:mm	3rd Angle Projection
Date	AUG/08/2014	Sheet:1/1	

Kiosk implementation Style guide

Pre-Commitment System

v0.2, 4th Feb 2015

State Government Victoria Department of Justice

How to use this document

This document describes the functionality and display of the Pre-Commitment site and kiosk touch screen keyboard. It is to be used as a reference document when developing and checking the deployed designs on the target hardware.

Units of measurement

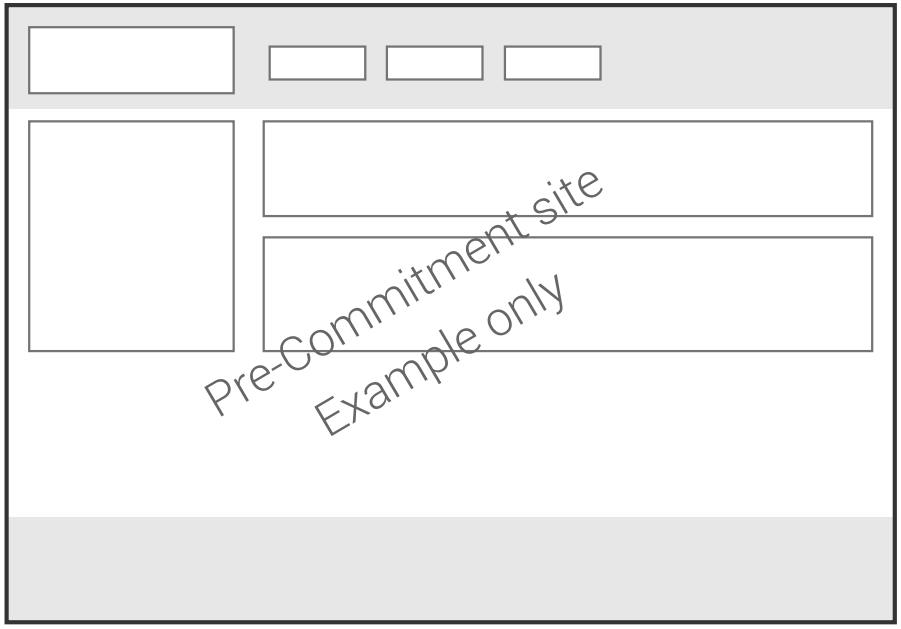
The touch screen keyboard has been designed to meet best-practice button and text sizes to support user interaction. While measurements can be supplied in pixels it is the final physical size on screen in millimetres that counts and all measurements have been stated as such. Pixel resolution varies from device to device. Ensure that the final physical sizes are validated on the actual deployed screen.

Photoshop artwork

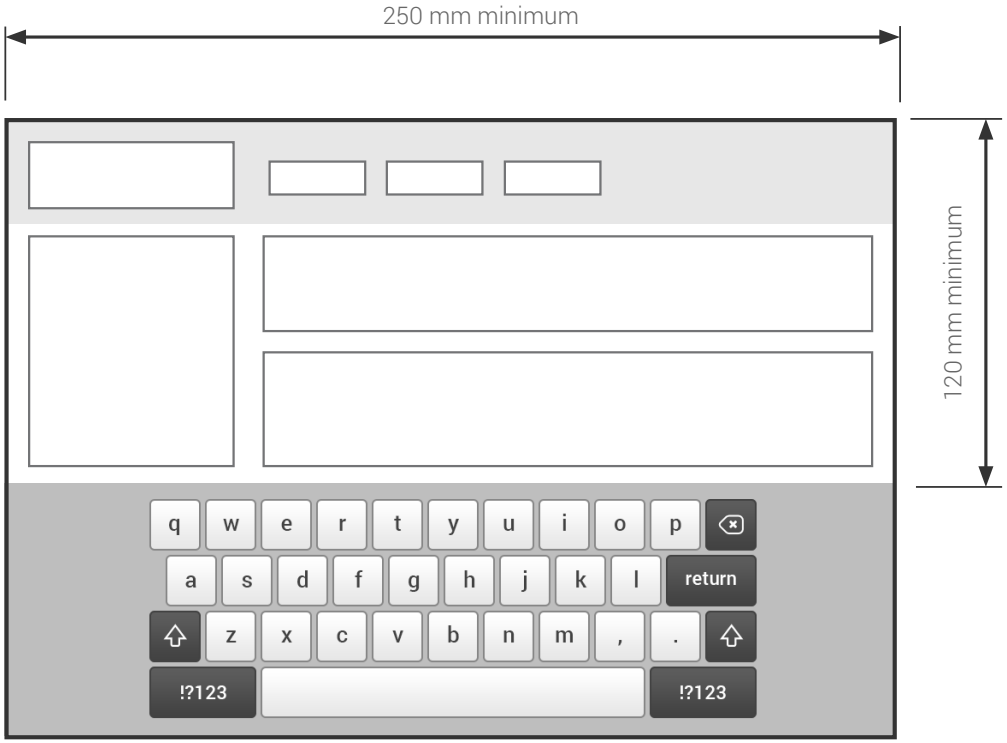
This document is to be used in conjunction with the supplied Photoshop files. Ensure that in deploying these designs that details such as alignment of components, size and colours are retained.

Use full size of screen to display site and keyboard

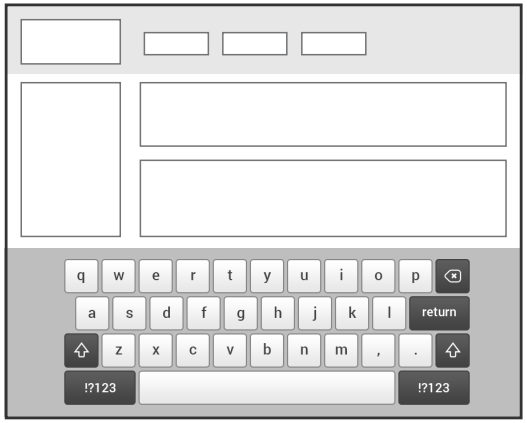
Do not show site in a smaller window

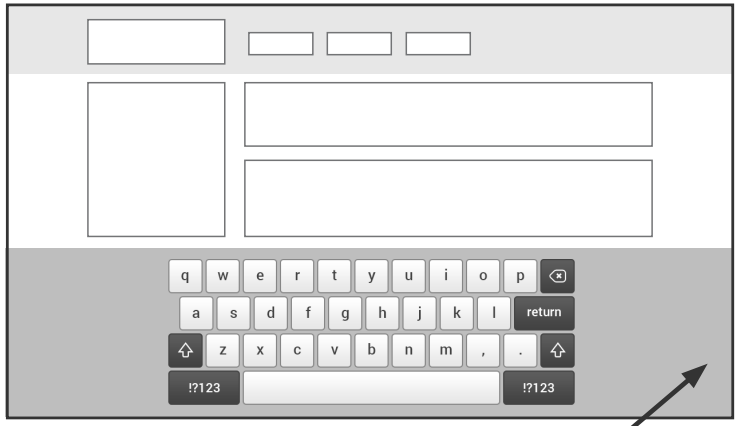


Minimum viewable area of site while keyboard is visible

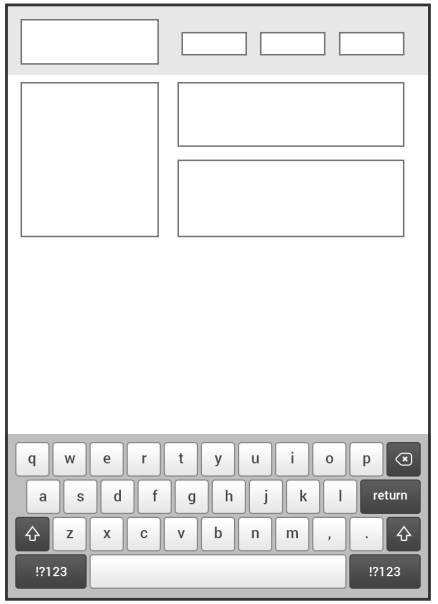


Correct arrangement of keyboard on varying screen sizes

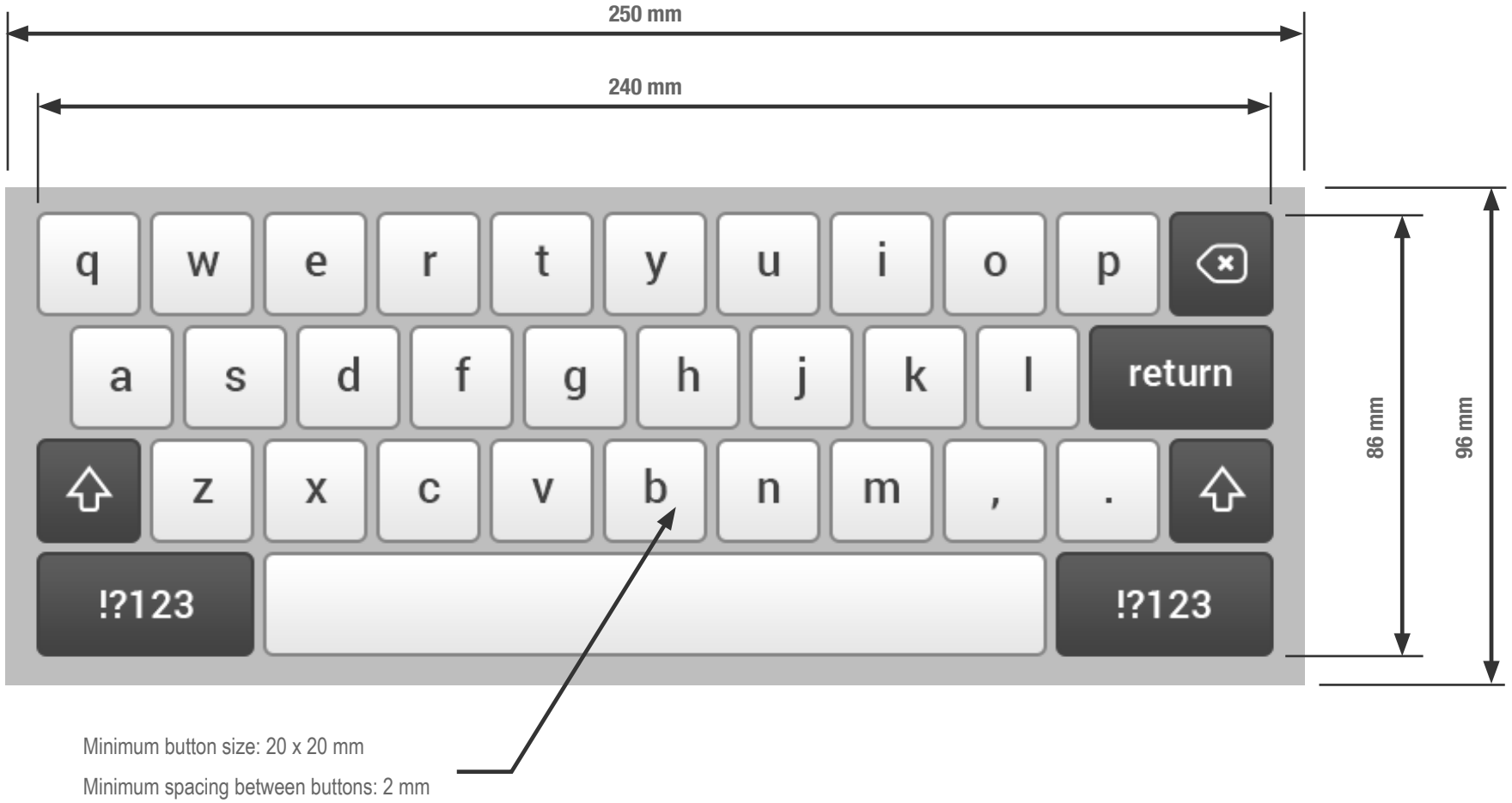




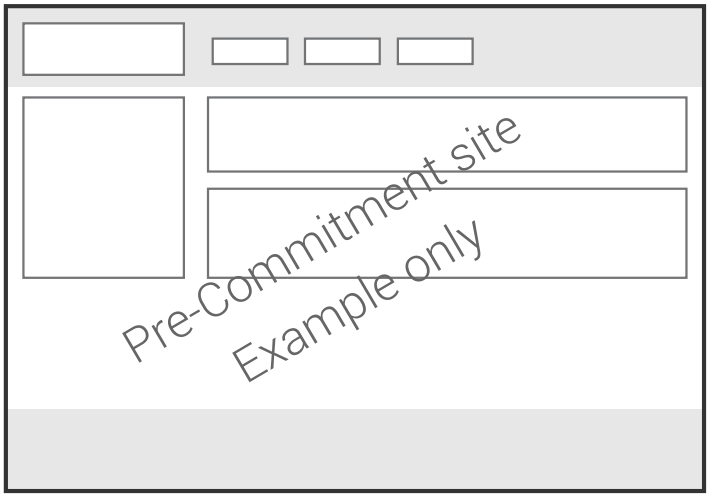
For larger displays extend grey background to edge of screen.



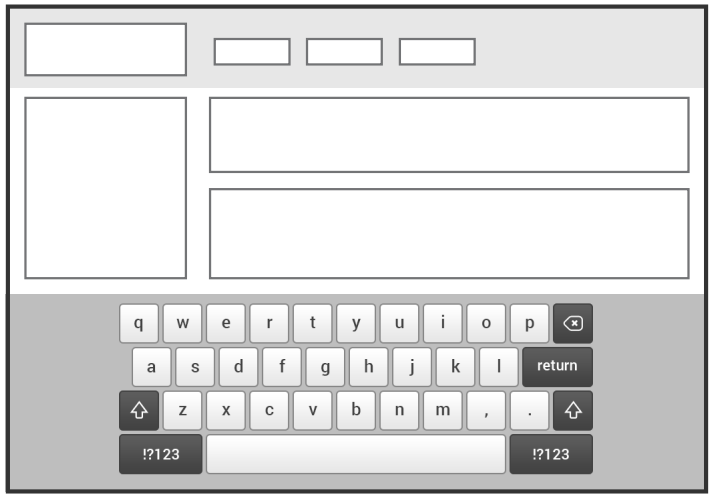
Minimum keyboard dimensions



Overlay behaviour of keyboard

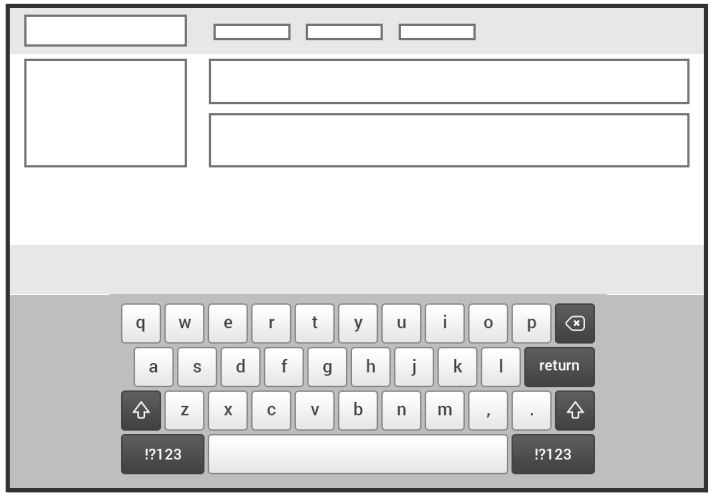


Preserve proportions of Pre-Commitment site when keyboard displays

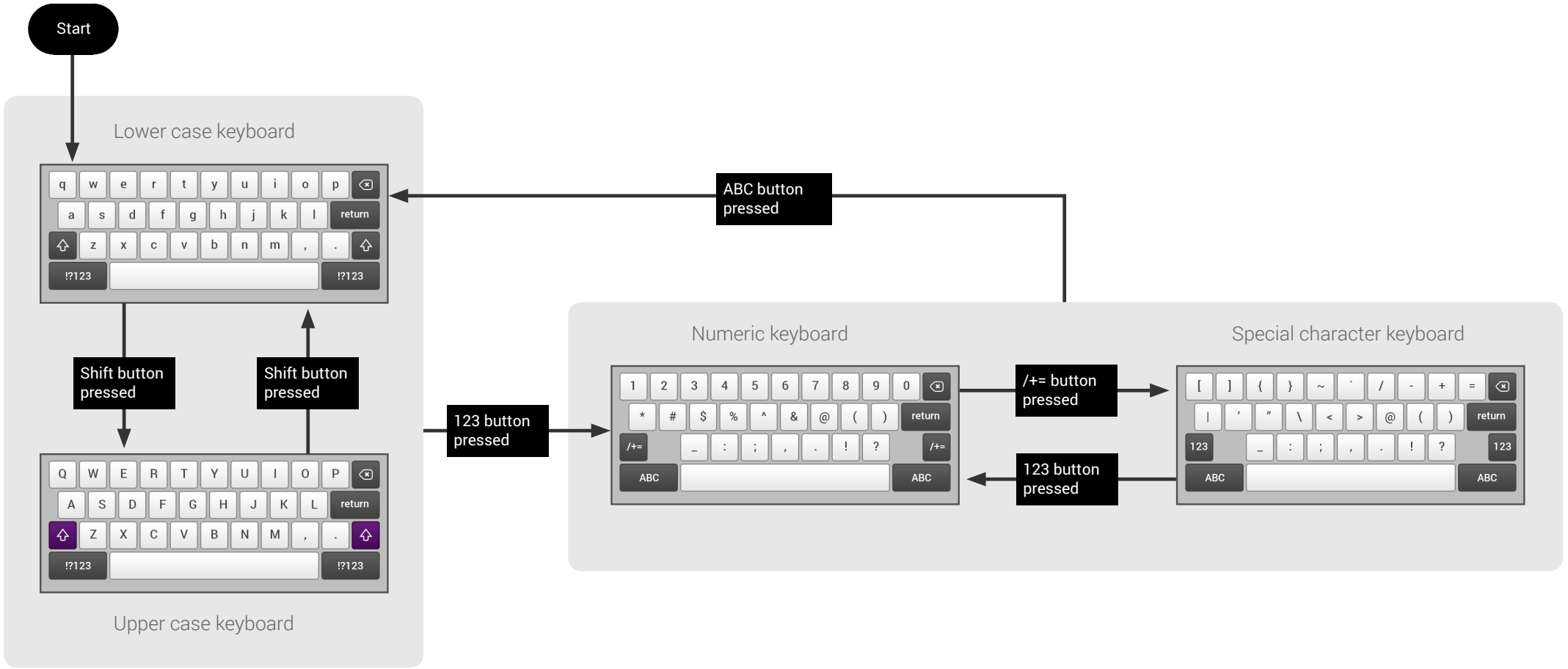


Keyboard overlays on top of underlying website.

Do not distort proportions of site



Keyboard toggle behaviour



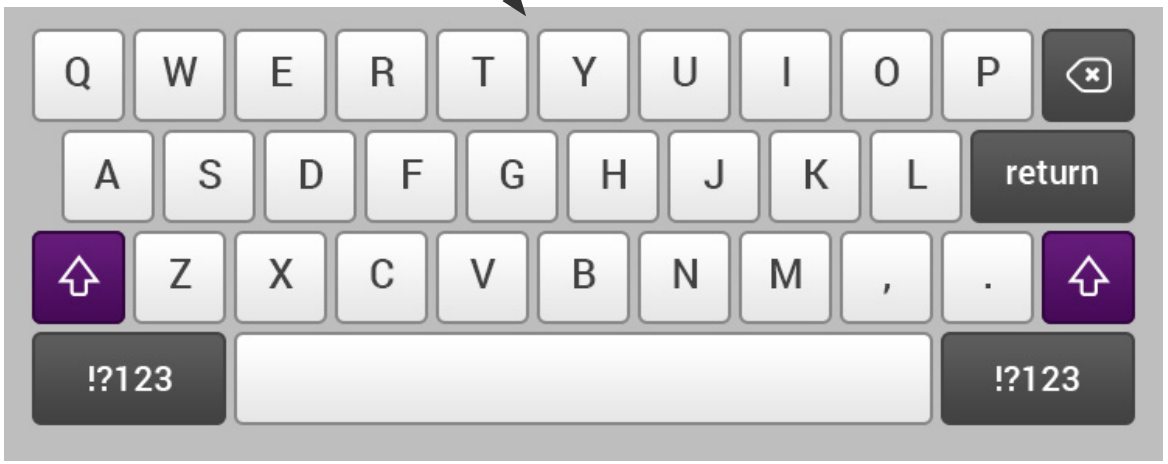
Standard button

Neutral	
Gradient from	#fff
Gradient to	#e7e7e7
Stroke	2px #868686
Text colour	#424242

Pressed	
Background	#d6d6d6
Stroke	2px #868686
Text colour	#424242

Keyboard background colour

#bebebe



Function button

Neutral	
Gradient from	#5d5d5d
Gradient to	#424242
Stroke	2px #424242
Text colour	#fff

Pressed	
Background	#333333
Stroke	2px #333333
Text colour	#fff

Toggle button (Shift button only)

Ensure both shift buttons are synchronised so that both respond when one is pressed

Neutral	
Gradient from	#5d5d5d
Gradient to	#424242
Stroke	2px #424242
Text colour	#fff

Pressed	
Background	#333333
Stroke	2px #333333
Text colour	#fff

On	
Gradient from	#470a59
Gradient to	#59196c
Stroke	2px #330540
Text colour	#fff

Pressed while on	
Background	#330540
Stroke	2px #330540
Text colour	#fff

Fonts and button labels

Roboto Medium

Roboto is a licence free font available at http://developer.android.com/design/style/typography.html

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890
!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~

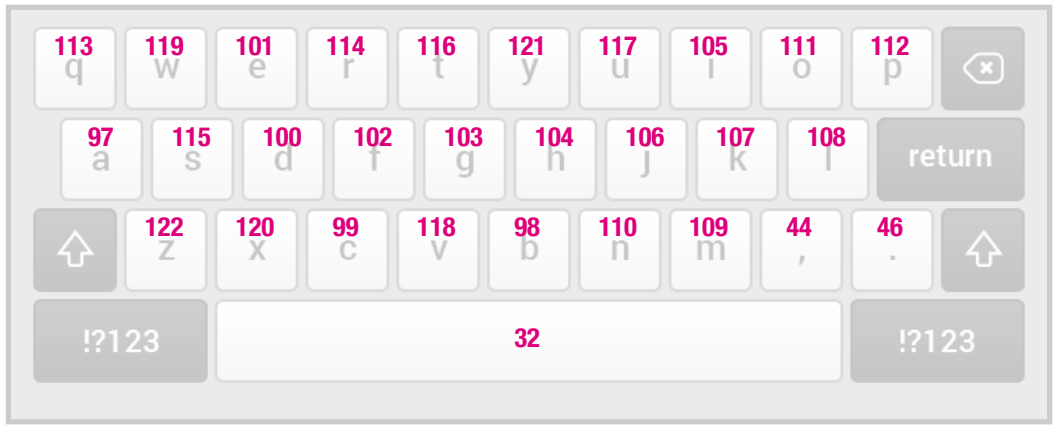
Shift and Backspace vector icons

The icons for the shift button and backspace button have been saved as individual vector artwork files and should be used in creating these buttons.

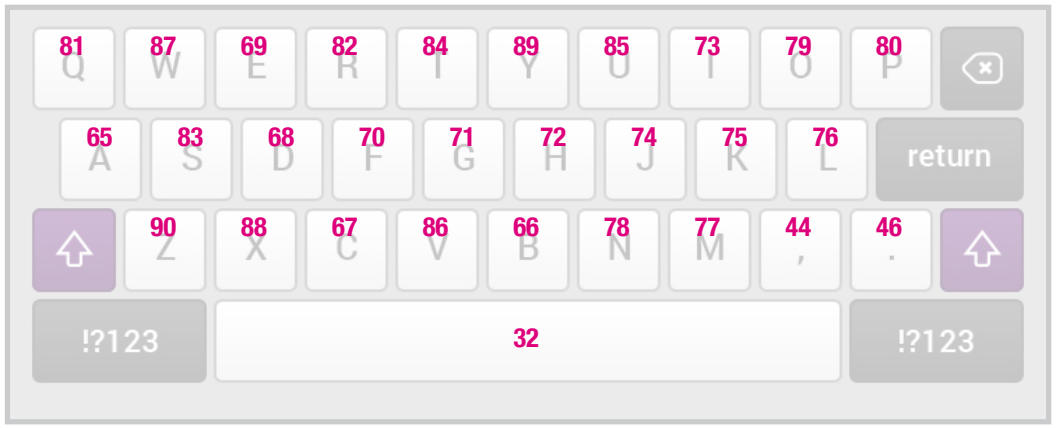
ASCII character code mapping

For further information refer to the 'ASCII printable characters' table at www.ascii-code.com

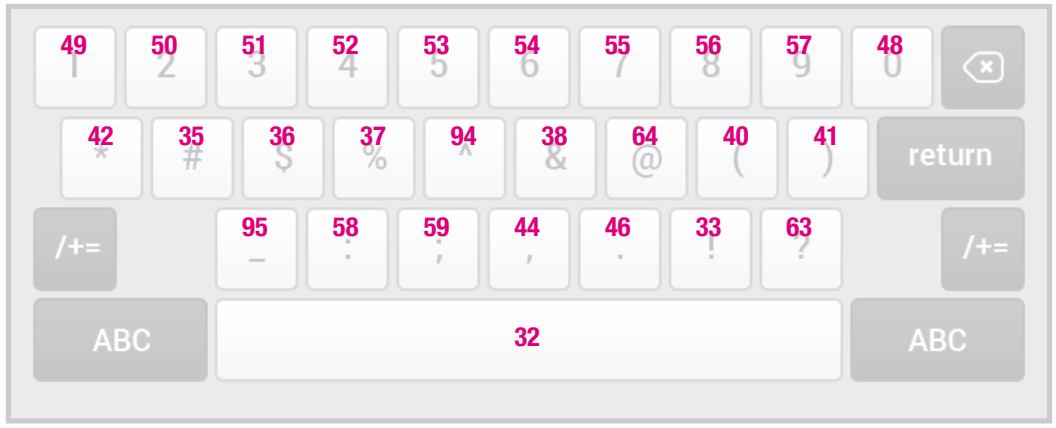
Lower case



Upper case



Numeric



Special characters