OORT WMD TDS

<Glance Home screen Feature>

Contact Information		
PM Owner	Roi Kim; Susan Hendrich	roikim; susanhe
Dev Owner	Heonmin Lim	limheonm
Test Owner	Jeongim Bae	v-jbae

Revision Summary			
Author	Date	Version	Comments
Jeongim Bae	4/14/2009	0.1	Initial Draft
Jeongim Bae	4/20/2009	0.2	Updated with test strategy
Jeongim Bae	4/23/2009	1.0	Updated with internal review and test case outline

Related Documents	
PM	http://mcpg/WDC/Crossbow/Specs/Glance%20Homescreen%20v2%20-%20Feature%20Spec.mht
Test	http://medpg/OORT/WMD/test/Test%20Design%20Specs/Glance%20Homescreen%20Framework%20TDS.docx
Dev	Development document, TRD's, etc
UE	User documents, manual references, error message files, etc.
Modeling	Threat modeling docs, etc.

1. Feature Description – Required for page 1

1.1 Overview

Glance Homescreen is a new Homescreen targeted for a consumer messaging user. This feature sorts incoming notifications from various sources by favorite contacts instead of applications, and displays them next to their associated contacts. So, we can see who the notifications are coming from in what applications.

The end user sets up 8 favorite contacts to sort notifications into and interacts with favorite contacts as well as associated plug-ins to see unattended notifications and handle them by invoking application specific actions such as reading unread SMS, or email messages, or returning missed phone calls.

Behind the scene, the feature keeps track of the favorite contacts that receive change notifications instead of polling for changes and maps direction keys, hot-keys, and speed dials for easier and quicker accessibility.

For native notifications (such as missed calls or unread SMS or emails), an associated plug-in finds matching contacts and then filters them by favorite contacts.

For non-native notifications such as MSN IM or Facebook, an associated plug-in may use partner APIs to sync up with changes, and optionally change the contact image to reflect the presence information such as available, busy, away, offline, etc.

The Glance home screen consists of the framework and plug-ins which are sample/reference code, but good enough to demonstrate the possibility of achieving a variety of goals.

2. Test Scope – Required for Page 1

2.1 What is Covered

Loading Glance Homescreen and non-Glance Homescreen

Navigation by the direction pad

- Navigation to each plug-ins (contact plug-in, application plug-in, and clock plug-in)
- Navigation to each contact square
- Navigation to base plug-ins (missed call, IM, unread text messages & emails)

Interacting with favorite contact plug-in

- Adding/Changing/Removing favorite contact
- Interacting with favorite contacts on the action screen.
- Invoking the speed dial function and hot key to dial assigned numbers for favorite contacts.

Changing Registry

- Changing D-pad mapping
- Changing Hot-key mapping

Testing will be done on only SP, USA and 320x240 resolutions

Sanity Testing for Title bar and Hardware key (Send, Home, Back, End key, SK1 and SK2)

2.2 What is not Covered

Testing on PPC is not covered as this is SP only feature, and any resolution beside 320x240 landscape mode is not covered.

Number of favorite contacts beyond 8

Variations of Base plug-ins (notification box1 = Missed IM, notification box2 = Unread Emails, notification box3 = missed call, notification box4 = missed txt message)

All registry possible combination for D-pad and Hot-key is not covered. We will cover basic user scenario.

Full test coverage meeting the RTM level of quality.

Base plug-ins and application plug-ins will be covered on each TDS.

We don't verify anything beyond Homescreen such as Start menu and title bar.

2.3 Dependencies and Integration Issues

How does this area relate to other pieces of the product? What pieces does it depend on? What pieces depend on it?

Non-native notifications such as MSN IM, Facebook, or MySpace have a dependency on those partners APIs.

2.4 Partnership

Consider the parties involved in the project. (e.g developer, core team, peer SDET, etc) What dependencies, if any, do we need to consider during our testing? For automation, how do your proposed changes/additions affect partner team(s)? Describe what measurements are in place to ensure safe/successful handoff to partner(s).

N/A

2.5 Risks/Complexity

What are the most risky areas? Why? Describe how you plan on accounting for and minimizing this risk in your testing.

Treating as a sample/reference code, we might miss some user scenarios.

UX and Security team is not involved at this point.

2.6 Key Scenarios

What are the key user scenarios for this feature that are required for shipping?

Katie sets up Jack as her favorite contact since he usually sends text and email messages to her. She sees unread text messages aligned with Jack's picture, selects his picture using the direction pad, and responds to his messages through the action screen.

Any communications from her favorite contacts such as miss call, txt message, email, IM are shown right next to their picture. Even Katie can reply to them in quick and easy way

2.7 Open Issues

Are there open issues regarding the feature? List the bug # and owner. If there is not a bug, then describe the open issues, owner, action item, and time frame for resolution.

- Supporting SIM contact for favorite contacts
- Removing/changing a favorite contact without deleting the contact in POOM DB
- Setting up the speed dial for applications such Calendar, Settings or Internet Explorer
- Is there default setting for D-pad and Hot-key?
- Is there a way to change favorite contact number for speed dialing?
- What happens to selection when one of keypad is pressed to wake the device in sleep mode, power cycling and unlocking?
- If user with in application presses Home key, what is expected Homescreen display?

3. TEST RESOURCES – REQUIRED FOR PAGE 1

3.1 RESOURCES

Describe the human resources required to test this feature for each of the areas below. What work can be outsourced or completed by a CSG? Use the table below to indicate owners and time estimates for each task.

All estimates should be expressed in **DAYS** and should be **Closed Door Estimates**.

PRE Code Complete

Pre CC								
#	Scheduled work items	Closed door estimate (days)			Scope of the work items changed	Actual time for work performed (days)		
		FTE	CSG	Vendor		FTE	CSG	Vendor
1	Dev, PM and TM spec reviews, research and assess test impact.							
2	Full/P1 TDS. Writing a series of test pass and test casing details.							
3	TDS review and make necessary revisions.							
4	WTT test case development. This includes writing complete repro steps in form of outsourcing quality.							
5	Ad hoc testing				Familiarize yourself with your feature and find design bugs near CC			
6	Test case review and make necessary revisions.							
7	Reserve Outsource resource				Create and submit outsource request for manual testing			
8	Lead CSG/Vendor overseeing their test status and progress (applicable to FTE only)							
	Total							
Automation								
9	Ramp up automation. Familiarize the unfamiliar code.							
10	Complete automation sections in TDS							
11	Code review UIAL from dev.							
12	BVT is written, reviewed and checked in. This is to pass BVT before dev declares CC.							
13	UI Automation to capture screen shots is written, reviewed, and checked in.							
14	Reserve Outsource resource				Create and submit outsource request for manual testing			
	Total							
	Sub Total							

POST Code Complete

Post CC								
#	Scheduled work items	Closed door estimate (days)			Scope of the work items changed	Actual time for work performed (days)		
		FTE	CSG	Vendor		FTE	CSG	Vendor
1	Weekly triage/status meeting							
2	Manual feature test pass on WWE key devices in primary screen orientation							
3	TDS/ADS/WTT update with regards to DCR, add/cut reviews, etc.							
6	Performance test pass (if applicable)							
7	App compatibility test pass (if applicable)							
8	Run AppVerifier and other test requirements per TDS							
9	Bug regressing/closing, adding/updating test cases throughout test passes (item 2 – 8)							
10	Ad hoc testing				Recommend 10-20% of time (4-8 hrs/week) spent finding/regressing bugs			
11	Lead CSG/Vendor overseeing their test status and progress (applicable to FTE only)							
	Outsourced Testing							
12	Sanity test pass on RUS and CHS (pilot)				Vendor to run P1 and P2 tests on non-USA Pilot Languages			
13	Legacy regression test pass on WWE. This includes regressing bug fixes in PS from previous release. Limit to those with high probably risk of regression.							
14	Dynamic Screen Rotation and secondary screen orientation on CEPC or device							
15	Verify test case exists for each fixed bug.							
16	IU Test pass							
	Total							
Automation								
17	Area Library (DATK) is written, reviewed and check-in.							

18	EC (DATK) is written, reviewed and checked in.							
19	POOM automation (if required) is written, reviewed and checked in.							
20	Cellcore automation (if required) is written, reviewed and checked in.							
21	UBOM automation (if required) is written, reviewed and checked in.							
22	Complete WMD peer code reviews (item 12 – 16)							
23	Complete external code reviews (item 12 – 16)							
24	All applicable automation is completed and running in the lab.							
	Outsourcing							
25	Complete Outsourceable Test Automation				Outsource the automation after EC area library and EC test cases are automated. (<10% area library, ~50% of test cases)			
26	Fix legacy automation failures							
	Total							
	Sub Total							

PRE Release Candidate

Pre RC								
#	Scheduled Work Items	Closed door estimate (days)			Scope of the work items changed	Actual time for work performed (days)		
		FTE	CSG	Vendor		FTE	CSG	Vendor
1	Regress/close bugs carried over from post CC. Need 0 bug counts to declare RC.							
2	IU bug bash, dogfooding							
3	RC EC Test Pass							
4	Lead CSG/Vendor overseeing their test status and progress (applicable to FTE only)							
	Total							
Automation								
5	Fix legacy automation failures from post CC							
6	Verify RTM quality lab automation results to get ready to hand-off to core team							
	Total							
	Sub Total							

POST Release Candidate

Post RC									
#	Scheduled Work Items	Closed door estimate (days)			Scope of the work items changed	Actual time for work performed (days)		
		FTE	CSG	Vendor		FTE	CSG	Vendor
1	Regress/close bugs carried over from pre RC (if any)							
2	Loc verification on RUS and CHS.							
3	Lead CSG/Vendor overseeing their test status and progress (applicable to FTE only)							
	Total							
Automation								
4	Close any outstanding automation bugs, verify RTM quality lab results to get ready to hand-off to core team							
5	Handoff maintenance of Automation to Vendor							
	Total							
	Sub Total							

Estimate Summary

	FTE	CSG	Outsource
Pre Code Complete			
Post Code Complete			
Pre RC			
Post RC			
Total			

	Total Manual Estimate	Total Automation Estimate
Pre Code Complete		
Post Code Complete		
Pre RC		
Post RC		

Summary		
Total (days)	**0**	**0**

3.2 HARDWARE

Describe the hardware resources required to test this feature. What device is this feature intended for? Does this feature require a special hardware configuration? What platforms must this feature be tested on? Are any special network configurations required?

CEPC support is required for daily automation. If the feature you are implementing can not be run in the lab environment or via automation explain the reasons. Describe any other hardware resources required to automate this feature.

Does the feature require or will be affected by different resolutions? What resolutions will testing cover?

Windows Mobile Standard 320x240 131dpi (Jackie or vCEPC SP)

Table of resolutions supported for Windows Mobile 6.0:

Device Type	DPI	Resolution	Dynamic Rotation?
~~PPC~~	~~96~~	~~240x240~~	~~-~~
~~PPC~~	~~96~~	~~240x320~~	~~Yes~~
~~PPC~~	~~128~~	~~320x320~~	~~-~~
~~PPC~~	~~192~~	~~480x480~~	~~-~~
~~PPC~~	~~192~~	~~480x640~~	~~Yes~~
~~PPC~~	~~192~~	~~480x800~~	~~Yes~~
~~SP~~	~~96~~	~~176x220~~	~~No~~
~~SP~~	~~131~~	~~240x320~~	~~Yes~~
~~SP~~	~~131~~	~~320x320~~	~~No~~
~~SP~~	~~131~~	~~440x240~~	~~No~~
~~SP~~	~~131~~	~~400x240~~	~~No~~
~~SP~~	~~131~~	~~240x400~~	~~Yes~~
SP	**131**	**320x240**	**No**

3.3 SOFTWARE

Describe any additional software resources required to test this feature. Are there any Fakeril, FakeWiFi, etc. changes needed? Any C++ API level cases needed?

Fakeril tool

3.4 TOOLS

Will you be using or developing any special tools? Who will be writing these tools (FTE, CSG, and Outsourced)? How long will it take to create the tools?

All estimates should be expressed in **DAYS** and should be **Closed Door Estimates**.

Tool Name	Closed Door Estimate	Owner (alias)	Notes
TOTAL Tools(days)			

3.5 AUTOMATION

Is automation necessary for this feature? Is so, is there existing automation for this area?

What is the automation strategy for this feature?

What level of automation will be completed for this feature?

Who is the core team owner for automation in this area?

N/A

3.6 BUG TRACKING

What is the product studio path where bugs should be logged?

What are the DevCode(s) associated with this feature?

Path: \Devices Platform\Shell\Homescreen

DevTeam: Samsung

3.7 ADDITIONAL NOTES

What additional test planning is required for this feature? Do new cases need to be added? Special regression testing?

[Registry setting information will be added here]

4. Detailed Test Strategy

4.1 Feature Details

Describe the design and architecture of this feature area? How does it work? How is it used? List and describe all interfaces, API's, and methods. Include as much detail as necessary to demonstrate understanding, document assumptions, and provide a basis for reviewing testing coverage.

The Glance Homescreen consists of plug-in DLLs and layout settings that reside on Registry. When this Homescreen layout is applied, the plug-in UI assets are placed on the screen according to the layout settings as soon as the plug-in DLLs are loaded.

The favorite contact plug-in presents favorite contacts that are assigned to native contacts. When the user chooses to take action, the favorite contact plug-in invokes its base plug-ins. The favorite contact reflects changes to its native contact as long as it remains associated, and becomes dissociated as soon as its native contact gets deleted from POOM DB.

The favorite contact plug-in is accompanied by the date/time plug-in and application plug-ins that also can be selected and invoked. The date/time plug-in presents the current date/time as long as it is loaded. The application plug-ins will present application specific notifications and/or custom actions without regard to contacts --- for example Email, Facebook or MySpace.

4.2 General Test Plan

Describe your overall approach to testing this area and your strategy for planning and implementing your test cases. What is the automation strategy for this feature? If you had to describe to a PM, developer, or a new tester what you are planning to test and how, what would you say?

We will create and execute manual test cases on four major areas in priority order as follows:

Glance Homescreen layout:

Navigation to each plug-in, to each contact square, and to each sub plug-in

Interacting with favorite contacts

Changing registry for D-pad and Hot-key

Automation won't be implemented for this testing.

Testing will be done only on

- Language : USA
- SP 320x240 131dpi by using Jackie (or vCEPC SP)

PPC:

	Radio		SKU		Resolution		Other
	GSM	**CDMA**	**Premium**	**Standard**	**Landscape**	**HiDPI**	
Build Equivalency Class #1	N/A	N/A	N/A	N/A	N/A	N/A	

Build Equivalency Class #2							
Build Equivalency Class #3							

SP:

	Radio		Platform		Resolution		Resolution
	GSM	**CDMA**	**QWERTY**	**Other**	**Landscape**	**HiDPI**	240x320
Build Equivalency Class #1	Full	N/A	N/A	N/A	Full	N/A	N/A
Build Equivalency Class #2							
Build Equivalency Class #3							

4.3 SETUP/INSTALLATION TESTING

Describe what setup needs to be tested and your approach. Consider the behavior of the feature after using "Image Update" from a previous release.

4.4 AREA BREAKDOWN

Describe how you will divide the area into manageable chunks (i.e. outline of areas and sub areas). Breakdown areas into order of priority.

Consider what automation areas will need to change. What directories will contain changes? What directories will be affected by the changes?

Loading Glance Homescreen and non-Glance Homescreen

Navigating by the direction pad

- Navigation to each plug-ins (contact plug-in, application plug-in, and clock plug-in)
- Navigation to each contact square
- Navigation to base plug-ins (missed call, IM, unread text messages & emails)
- Changing registry key for D-pad

Interacting with favorite contact plug-in

- Interacting with favorite contacts on the Homescreen:
    - Presenting favorite contacts with pictures or first names (unless unassigned).
    - Adding/changing/removing favorite contacts

- Interacting with favorite contacts on the action screen:
    - Presenting Action screen
    - Changing a picture and bringing up [contact card] screen.
    - Invoking base plug-in applications when a direction key pressed.
- Selecting favorite contacts through Hot keys or speed dialing
    - Selecting favorite contacts by means of Hot-keys.
    - Speed dialing to the favorite contacts
    - Changing registry for Hot-key

4.5 API TESTING

Are there any APIs that require automation as part of this feature? If so, list and describe the API.

Explain how you will automate the API testing. What language? How will the daily automation execute the API test cases?

N/A

4.6 AUTOMATION AREA LIBRARY CHANGES

Are there any existing Area Library methods being changed? If so, list and describe.

What Area Library references need to be added to current Area Library code? Why?

List and describe any new methods that will be added to the Area Library, including method signatures.

Are there existing Area Library methods (possibly in a different Area Library) that perform the same or similar operation?

Does your feature have new UI? Be sure to add test cases which take screen captures for the localization test pass.

N/A

4.7 ERROR HANDLING/FAULT TOLERANCE

What methods will you use to invoke error testing? What is the expected behavior?

N/A

4.8 IMAGE UPDATE (IU) IMPACT

The image update (IU) technology is used by many OEMS to update phones in the field with service packs based on fixes from Microsoft and their own platform. Please assess the testing needed for IU scenarios based on the design of this feature by answering the questions below.

- Does the design of this feature require any changes that impact public APIs used by external /internal apps. If so, please describe.
- Will the design affect the database schema? Will any changes affect the way user data is stored?
- Will any new registry keys be added, removed, or moved? Will any registry keys be repurposed for new functionality?

Describe the test approach for verifying that all the scenarios above are handled effectively during an IU update from a previous milestone to the current milestone. Include verification steps for ensuring the IU update was successful and the feature functions as expected. Are there any test scenarios that require additional tools?

No. It doesn't have IU impact at this point since this is sample/reference code.

4.9 PERFORMANCE

Are there any performance requirements for this feature? What areas do you plan to investigate or target with benchmark tests, profiling, etc. in the Nexus timeframe? Is boot time affected by this feature?

We will measure the time it takes to boot with Glance Homescreen and to change Glance Homescreen from/to another Homescreen layout.

4.10 STRESS

Describe your strategy for stress testing your component. Will you test low/no memory, disk space scenarios?

If targeted MTTF should be run on the component - what exe/area will you target?

Are there any modifications you can make to the device to ensure better coverage of your component during MTTF? (ie: pre-populating the device with docs/music/photos/email etc?)

Are there any xHopper parameters that will ensure better coverage of your feature? (ie: rotation, Incoming SMS/Voicemail/PhoneCall, Launching an exe every so often?)

MTTF will be run for Stress testing

4.11 LEAK

Describe your strategy for Resource Leak testing your component.

How can you find Memory leaks/Handle leaks/etc?

App Verifier should be run on the code prior to code complete. Are there any concerns for this feature?

App Verifier will be used to make sure there is no leak.

4.12 BATTERY

Does this feature have any implications on power consumption? If so, why?

The Glance Homescreen has more notification (data connection) to use battery life will be impacted. If resources exist, we will run a DoU on Jackie with Sliding panel Homescreen layout to create a baseline and then run a DoU with Glance Homescreen layout.

It shouldn't use battery more than 10%.

4.13 INTERNATIONAL/LOCALIZATION

How will you ensure that your area will work correctly for International markets? Address International Sufficiency or "Z" issues (date/time, currency formats, sort order, etc.), as well as use of extended, Unicode, and/or DBCS characters. Be sure to include the pilot languages in your testing. Pilot language breakdown: 60% English, 20% Fareast, 20% European

No. This feature only supports for only USA build.

4.14 SECURITY

Are there any Security issues that are concerns for this feature?
Should file fuzz testing be done for this feature? If any file is accessed or used in this feature, then file fuzzing is required per Microsoft security.

File Fuzzing Tool Sites:
http://medpg/mese/sdl/Pages/MeseSecTools.aspx
http://seciis/sites/sec/tools/default.aspx

N/A

4.15 LOGO TEST KIT

Logo Test Kit is a suite of both manual and automated tests created by a Microsoft team and given to OEMs to run on their devices. All their products need to pass these tests before they can sell their devices include Microsoft product software on their products. For any major AKU release, the WMD team runs a LTK pass as part of their EC criteria. The goal of the test run is to identify code defects or regressions that could cause an inaccurate test case failure on OEM devices. This pass ensures each AKU can pass the logo test and thus saves time and money for both Microsoft and our OEMs. Should the feature be added to the LTK? Why?

Unknown

4.16 RELEASE CRITERIA

What scenarios need to be part of the milestone exit/release criteria? Are there specific metrics that must be met?

The release criterion is all test cases pass for test cases with P1, P2 priority.

4.17 DOCUMENTATION

Is there any documentation associated with this component? How will you verify its accuracy? List the location of the documentation

5. TEST CASE OUTLINE

Outline the test cases covered for each area of the feature. Provide a link to the test cases in the WTT Datastore.

$\Crossbow_OORT\Shell\Today/Home\Glance Homescreen\Framework

6. FUTURE RELEASE ISSUES

List issues or bugs, which need to be resolved or revisited for the next version of the product. This is also good place to describe any ideas you have for future testing approaches, strategies, tool ideas, etc.

7. GLOSSARY & ACRONYMS

Term/Acronym	Definition

Test Case Outline and Titles

Loading Glance Homescreen layout and non-Glance Homescreen layout

- P1. Verify Glance Homescreen can be loaded after selecting "Glance Homescreen" layout on display setting
- P1. Verify Non-Glance Homescreen can be loaded after selecting non-Glance Homescreen such as "Sliding panel" layout on display setting
- P2. Verify Contact plug-in is placed on top, Application plug-in on bottom left, and Clock plug-in on bottom right on the Homescreen.
- P2. Verify Contact plug-in is consisted of 8 contact squares
- P3. Verify each contact square displays "+ add contact" image
- P3. Verify there is no highlighted selection right after loading Glance Homescreen
- P3. Verify there is no highlighted selection right after unlocking the device
- P3. Verify background image can be applied after changing a background on Display setting.
- P3. Verify color scheme can be applied after changing a color scheme on Display setting.
- P3. Verify Glance Homescreen is still displayed after Power Cycling
- P3. Verify Glance Homescreen is still displayed after Soft Reset

Navigating by means of the direction pad

- P1. Verify selection moves to each contact square
- P1. Verify selection moves to each application slot
- P1. Verify selection moves to clock plug-in
- P3. Verify selection moves when D-pad navigation moves up
- P3. Verify selection moves when D-pad navigation moves down
- P3. Verify selection moves when D-pad navigation moves left
- P3. Verify selection moves when D-pad navigation moves right
- P2. Verify first contact square is highlighted by pressing any D-pad key after booting or unlocking the device
- P3. Verify Glance Homescreen can be navigated while calling
- P3. Verify Glance Homescreen can be navigated after unlocking the device
- P3. Verify selection returns to where it was when a phone call is ended (e.g. action screen)
- P3. Verify selection returns to where it was when appointment reminder is dismissed

Changing registry for D-pad

- P3. Verify D-pad navigates as changed after registry value for contact square 1 is changed

- P3. Verify D-pad navigates as changed after registry value for contact square 2 is changed
- P3. Verify D-pad navigates as changed after registry value for contact square 3 is changed
- P3. Verify D-pad navigates as changed after registry value for contact square 4 is changed
- P3. Verify D-pad navigates as changed after registry value for contact square 5 is changed
- P3. Verify D-pad navigates as changed after registry value for contact square 6 is changed
- P3. Verify D-pad navigates as changed after registry value for contact square 7 is changed
- P3. Verify D-pad navigates as changed after registry value for contact square 8 is changed
- P3. Verify D-pad navigates as changed after the registry value for application square 1 is changed.
- P3. Verify D-pad navigates as changed after the registry value for application square 2 is changed.
- P3. Verify D-pad navigates as changed after the registry value for application square 3 is changed.
- P3. Verify Clock navigates as changes after the registry value for Clock is changed.

Interacting with favorite contact plug-in:

- Interacting with favorite contacts on the Homescreen:
    - Presenting favorite contacts with pictures or first names (unless unassigned).
        - P1. Verify a favorite contact's picture is displayed at the image in the middle of contact square when an Outlook contact with a picture is assigned.
        - P1. Verify a favorite contact's first name is displayed at the image in the middle of contact square when an Outlook contact without a picture is assigned.
        - P2. Verify a favorite contact's first name is displayed at the image in the middle of contact square when a SIM contact is assigned.
        - P3. Verify "unnamed" is displayed at the image in the middle of contact square when contact has no name and no picture.
        - P2. Verify a changed picture is displayed for a favorite contact when the picture of a contact is changed on Action screen.
        - P2. Verify a changed picture is displayed for a favorite contact when the picture of a contact is changed from Contact in POOM DB.
        - P3. Verify a favorite contact's first name is replaced by its picture when the picture of a contact is assigned.

        - P3. Verify a favorite contact's picture is replaced by its first name when the picture of an Outlook contact becomes unassigned.
        - P3. Verify changed first name is displayed for a favorite contact when the fist name of an Outlook contact is changed while the picture remains unassigned.
        - P3. Verify "+add contact" image is displayed as soon as a favorite contact gets deleted from Contact.
        - P3. Verify "+add contact" image is displayed as soon as a favorite contact gets removed.
    - Setting up favorite contacts (adding/changing/removing favorite contacts).
        - P1. Verify [Contacts Creation] screen is brought up in case there are no existing Outlook/SIM contacts while newly adding a favorite contact.
        - P2. Verify [Contacts] screen is brought up in case there are existing Outlook/SIM contacts while newly adding a favorite contact, where the user can select one of them.
        - P2. Verify [Contacts] screen is brought up in case there are existing Outlook/SIM contacts while changing an existing favorite contact.
        - P2. Verify "+add contact" image is displayed if the user removes an existing favorite contact
        - P3. Verify each favorite contact can be added and deleted.
        - P3. Verify a favorite contact can be assigned to the existing favorite contact and removed from old slot.
        - P3. Verify a favorite contact is replaced by another existing favorite contact.
        - P3. Verify first, fourth and last favorite contact can be added and changed.
- Interacting with favorite contacts on the action screen
    - P3. Verify "Choose an action…" is displayed at the top left on the action screen.
    - P3. Verify contact square is displayed as bigger images with notification icons
    - P3. Verify each notification icon has a highlighted arrow indicating the direction of base plug-in
    - P3. Verify the first name is displayed below contact square.
    - P2. Verify "Change Photo" is displayed for LSK and "Cancel" for RSK
    - P2. Verify [select-a-picture] screen is displayed when "Change Photo" is pressed.
    - P2. Verify It brings back to the favorite contact with highlighted on Homescreen when "Cancel" is pressed
    - P2. Verify It brings back to the favorite contact with highlighted on Homescreen when "Back" button is pressed
    - P2. Verify [contact card] screen is displayed when "Select" button is pressed on a favorite contact of an Outlook contact.
    - P3. Verify [SIM contact card] screen is displayed when "Select" button is pressed on a favorite contact of a SIM contact.

- P2. Verify base plug-in application is invoked when [↑] button is pressed on a favorite contact; e.g. Reading unread IM messages
- P2. Verify base plug-in application is invoked when [→] button is pressed on a favorite contact, e.g. Reading unread email messages.
- P2. Verify base plug-in application is invoked when [←] button is pressed on a favorite contact, e.g. Reading unread SMS messages.
- P2. Verify base plug-in application is invoked when [↓] button is pressed on a favorite contact, e.g. Returning missed phone calls.

- Selecting favorite contacts through Hot-keys and speed dialing
  - Selecting favorite contacts by means of Hot-keys. (I assumed number '1' to '8' is assigned for the contact square by default)
    - P2. Verify Hot-keys '1' to '8' bring up [Action screen] of the favorite contact selection when a Hot-key is pressed and held for 1 second
    - P3. Verify nothing happens when one of mapped Hot-key is pressed and held if the favorite contact remains unassigned
    - P3. Verify nothing happens when unmapped Hot-key is pressed and held.
    - P3. Verify Glance Homescreen updates the favorite contact order if speed dial setting is changed on non-Glance Homescreen.
  - Speed dialing to the favorite contacts
    - P2. Verify a speed dial number can be selected if speed dial is being used for the first time on that contact in case the contact has multiple phone numbers.
    - P2. Verify [speed dialing] screen is brought up when the Hot-key is held down continuously from Homescreen
    - P2. Verify [speed dialing] screen is brought up when the Hot-key is pressed and held on Action screen
    - P3. Verify the text "Calling [first name of contact] in [seconds remaining] seconds" is displayed at the bottom on Action screen.
    - P3. Verify the user remains in the action screen if Hot-key is released before the countdown timer reaches at 0.
    - P3. Verify a call is placed when countdown timer reaches at 0
    - P3. Verify the user is returned to the Homescreen with nothing in focus once the call is ended.
    - P3. Verify "Open Speed Dial" menu item is disabled from the call dialer window.
    - P3. Verify "Add to Speed Dial" menu item menu item is disabled from the contact card.
    - P3. Verify "Edit to Speed Dial" menu item is disabled from the contact card menu item.
    - P3. Verify "Open Speed Dial" menu item is enabled from the call dialer window on non-Glance Homescreen

- P3. Verify "Open Speed Dial" menu item is enabled from the call dialer window on non-Glance Homescreen
- P3. Verify "Open Speed Dial" menu item is enabled from the call dialer window on non-Glance Homescreen
- P3. Verify application cannot be assigned to speed dials '1' to '8'.
- P3. Verify a favorite contact reflexes change when user changes speed dial settings for favorite contact on non-Glance Homescreen
- P3. Verify the contact is not displayed on Glance Homescreen if the user set the speed dial setting to lager number than 8 on non-Glance Homescreen.

- Changing registry for Hot-key
  - P3. Verify Hot-keys navigates to and speed dial to Contact 1 after keymapping registry value for Contact 1 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 2 after keymapping registry value for Contact 2 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 3 after keymapping registry value for Contact 3 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 4 after keymapping registry value for Contact 4 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 5 after keymapping registry value for Contact 5 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 6 after keymapping registry value for Contact 6 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 7 after keymapping registry value for Contact 7 is changed
  - P3. Verify Hot-keys navigates to and speed dial to Contact 8 after keymapping registry value for Contact 7 is changed
  - P3. Verify multiple Hot-keys can be mapped to a contact square.
  - P3. Verify "second remaining" reflects change after "SpeedDialCoutDownSeed" registry value is changed