class host.presentation

---

presentation::WaitForOtherPhoneActivity

+ cancelClicked(View) : void
+ handleState(ServiceState) : void
# onCreate(Bundle) : void
# onServiceConnected() : void
+ sendClicked(View) : void
+ stateChanged(ServiceState) : void

---

presentation::ConnectedActivity

handleState(ServiceState) : void
onCreate(Bundle) : void
onServiceConnected() : void
+ quitClicked(View) : void
+ stateChanged(ServiceState) : void

---

presentation::UnconnectedActivity

handleState(ServiceState) : void
onCreate(Bundle) : void
onServiceConnected() : void
+ stateChanged(ServiceState) : void

Activity
presentation::AbstractHostActivity

boundService: IHostService
- dialog: ProgressDialog
- isBound: boolean
LOG_TAG: String = getClass().getS... {readOnly}
- mConnection: ServiceConnection = new ServiceConn...
- menu_id: int = R.menu.menu
- noBinding: boolean = false
RESULT_USER_EXIT: int = 42 {readOnly}

completeExit() : void
~ doBindService() : void
~ doUnbindService() : void
+ onActivityResult(int, int, Intent) : void
+ onCreateOptionsMenu(Menu) : boolean
+ onOptionsItemSelected(MenuItem) : boolean
+ onPause() : void
+ onResume() : void
# *onServiceConnected() : void*
- removeServiceBinding() : void
- showAbout() : void
showErrorDialog() : void
- showSettings() : void
+ *stateChanged(ServiceState) : void*

«interface»
controller::IHostServiceListener

+ stateChanged(ServiceState) : void

-serviceListener

«interface»
controller::IHostService

+ getConnectionDetails() : ConnectionParameter
+ getErrorMsg() : String
+ getState() : ServiceState
+ setListener(IHostServiceListener) : void

#boundService

Service
controller::HostService

- accessoryConnector: IAccessoryConnector
+ ACTION_USB_ACCESSORY_ATTACHED: String = "to.sven.androi... {readOnly}
- binder: IBinder = new HostBinder() {readOnly}
- connectorListener: IAccessoryConnectorListener = new IAccessoryC... {readOnly}
- dependencyContainer: IHostDependencyContainer
- errorMsg: String
- hostLogicListener: IHostLogicListener = new IHostLogicL... {readOnly}
- LOG_TAG: String = "HostService" {readOnly}
- logic: ILogic
- notification: HostServiceNotification
- serviceListener: IHostServiceListener
- socketConnector: ISocketConnector
- socketConnectorListener: ISocketConnectorListener = new ISocketConn... {readOnly}
- state: ServiceState = ServiceState.NO...

+ changeState(ServiceState) : void
- executeSocketConnector() : void
+ getConnectionDetails() : ConnectionParameter
+ getErrorMsg() : String
+ getState() : ServiceState
+ onBind(Intent) : IBinder
+ onCreate() : void
+ onDestroy() : void
+ onStartCommand(Intent, int, int) : int
+ onUnbind(Intent) : boolean
+ setListener(IHostServiceListener) : void

---

controller::HostServiceNotification

- context: Context {readOnly}
- notification: Notification
- NOTIFICATION_ID: int = R.string.host_s...
- notificationManager: NotificationManager

+ cancel() : void
+ changeNotification(int) : void
+ HostServiceNotification(Context)
+ stateChanged(ServiceState) : void

Activity
presentation::AccessoryAttachedActivity

- LOG_TAG: String = "AccessoryAttac... {readOnly}

onCreate(Bundle) : void

PreferenceActivity
presentation::HostPreferencesActivity

onCreate(Bundle) : void

-state

«enumeration»
controller::ServiceState

NOT_CONNECTED
WAIT_FOR_OTHER_PHONE
CONNECTED
CONNECTION_LOST
ERROR

-notification