BuzzerStar V2.1.4 – Screen Anforderungsprofil

2013-9-8 / v2.1.4 / Enger

Allgemein: Jeglicher Text in der App gehört in ein XML Sprachfile. Sauberer Code, Kommentare in English, Dokumentation in DE/EN.

Neu:

- Punkt Erweiterungen

Allgmein:

1. Man kann durch die Screens wischen von rechts nach links und links nach rechts
2. Alle 3 Tage prüfung auf updates (http head anfrage vom handy an www.buzzerstar.com/app.php (siehe xml configuration): Last Modified???
    a. Zwangsupdate wenn neues update vorhanden
        i. Installiere App von www.buzzerstar.com/app.php
3. Jeder der aufgelisteten Screens kann via xml configuration aktiviert oder deaktiviert werden
4. Die xml configuration ist eine xml steuerdatei, die der apk beiliegt und das verhalten der app steuert, will man einen xml wert ändern, so muss eine neue app ausgeliefert und kompiliert werden
5. Funktionalität muss auf Smartphones vom Typ Android jeglicher Bildschirmauflösung, sowie Android Tablets laufen, Ziel: lauffähig ab Android Version 2.2/2.3

BootScreen:

- Blende Zentral BuzzerStar Logo ein
- Blende darunter eine: www.werbung.com (siehe xml configuration)
- Warte 4 Sekunden (Schreibe: Initialisiere Crypto Handler, Initialisiere SSL-256 Bit Verbindung, Inititlisiere AES-256 Bit Verschlüsselung, Crypto Setup fertig-danach warte weitere 1 Sekunden und zeige Start Screen) -> bitte ins XML Sprach file (DE,EN)

Start Screen:

1. Suchfeld incls such button muss quer und hochformat auf einem screen lesbar sein
2. Suchfeld mit Hintergrund „Kinofilme, Serien, Anime, Pornos“
3. Klick ins Suchfeld löscht Eintrag: „Kinofilme, Serien, Anime, Pornos“
4. Suchfeld mit X Button – löscht Suchfeld
5. Button: Suchen -> Sendet suche an Proxy Server
6. Suchergebnisse darstellen: oben Suchfeld mit aktuellem Suchbegriff, darunter 50 Ergebnisse, jeweils mit Kategorie Bildchen, rechts davon „Download This“-Icon
    a. Neu: Klick auf Such Ergebnis Startet Streaming im Stream Fenster

Stream Screen:

1. Bei Wechsel ins Stream Feld: Stream Starten im Querformat
2. Lautstärkeregeler
3. Idealfall: Features des MX Players(https://play.google.com/store/apps/details?id=com.mxtech.videoplayer.ad&hl=de)
4. Balken mit aktueller Position des Streams (klicken und schieben um vor oder rück zu spuhlen)
5. Fullscreen Video anzeigen
6. Beenden der App oder Drücken des Home buttons: Stoppen des Streams an Position X
    a. Merken der aktuellen Position X und bei Handy neustart/App neustart/wechsel auf Stream screen wieder an Position X starten
7. Download Stream button -> lädt im Hintergrund im Download Screen diesen Film runter(wenn mehr als X aktive Downloads dann ans Ende der Download Queue, sonst starte Download mit niedrigster Prioriät)
8. Lautstärkerregeleer, Streamposition balken und Download Stream nach 10 sek inaktivität ausblenden-> dann fullscreen video
9. Traffic Priorität: Stream: 75% der Handyverbindung für Stream nutzen
10. Beenden der App: Stream position speicher, sodass man bei neustart dort fortsetzen kann, stop stream
11. Ausblenden der App: active streams position speichern, stop stream
12. Ausstellen, das der Bildschirm dunkel wird

Download Screen:

1. Max 3 Downloads parallel(via xml config, später mehr)
2. Schreibe oben über den Downloads (xml sprachfile): **Deine Downloads und Streams werden durch eine abhörsichere 256 Bit SSL und 256 Bit AES Verbindung geschützt.**
3. Für jeden aktiven Download: Prozentanzeige (Gauge) fertig, XYZ MByte geladen, ZXY Mbyte verbleiben, Prognose: noch XZY Minuten downloaden-> Download Position speichern, nach jedem Download Chunk
4. Fall: keine Internetverbindung: n=1..300, warten: s=(n*2)+n Sekunden, danach wieder Download Request stellen.
5. Wenn n=300 und kein download resume möglich, dann speichere für jeden aktiven Download die position und schreib nachricht(language xml): „Dein Handy hat kein Internet. Die App lädt deine Filme weiter, wenn du wieder Internet auf deinem Smartphone hast.
6. Nach dem n=300 alle 60 sekunden prüfen ob Internetverbindung
    a. Ja: Download Resume
    b. Nein: hibernate
7. Starten der App: Download Resume
    a. Stream und Download: 75% Stream Traffic und 25 % Download Traffic
13. Traffic Priorität:

    a. Stream und Download -> Stream 75% / Download: 25% der Handyverbindung für App benutzen
    b. **Nur Download:** 100% der Handyverbindung für Download nutzen
8. Beenden der App: active Downloads speichern, Download Queue speichern, download stoppen
9. Ausblenden der App: active Downloads speichern, Download Queue speichern, download fortführen

News Screen:

1. Screen zweiteilen (oben: neue Kinofilme) unten: neue Serien
2. Beim ersten Klicken oder ersten Wechsel auf News Screen Tab -> Stelle News Request an Server und parse Result XML
3. News Result Request für n(=aktuell gleich 5) Tage cachen auf dem Handy, erst danach wieder neuen News Request senden
4. Aufbau: Kinofilm/Serien für jeden Eintrag:
    a. Position (Bsp: 1..n;n=10)
    b. Bild (Bsp: 

)
    c. IMDB (Bsp: 7.5)
    d. Summary (Bsp für Breaking Bad: „Krebskranker Chemielehrer kocht Crystal“)
    e. Name (Bsp: Breaking Bad der KinoFilm)
        i. => Ein klick auf Position, Bild, Imdb, Summary oder Name Startet Buzzerstar Suche nach „Name“
5. Aktualisierung des News Screen
    a. Hardcodiert in App: alle 5 Tage

Friend Screen:

1. Oben text: „Klicke auf das Bild und Wähle deine Freunde aus, die du über BuzzerStar Benachrichten willst“ -> ins XML Sprachfile

2. Bild:
3. Ein Klick auf dieses Bild: öffnet Textfeldpopup mit den Gesamten Telefonbuch (scrollbar) des Handys
4. N A C H jedem Namen des Telefonbuchs ist ein möglicher Haken zu setzen(wenn man ihn setzt, bekommt dieser Eintrag eine SMS)

5. Button Senden (Klick darauf Sendet SMS mit Text an alle ausgewählten Handynummern: ***„Hallo $NAME_AUS_TELEFONBUCH, unter www.buzzerstar.com kannst du dir kostenlos Kinofilme, Serien, Animes und Kinderfilme auf dein Handy downloaden.“*** -> XML SPRACHFILE: Wichtig, $NAME_AUS_TELEFONBUCH so zu kürzen/begrenzen, dass SMS MAXIMAL 160 Zeichen lang ist.
6. Nach dem erfolgreichen Sendevorgang: Text: „Vielen Dank, wir haben deine Freunde benachrichtigt.“->XML Sprachfile

History Screen:

1. 2 geteilter Screen: oben Streams/ unten Download
2. Stream feld: überschrift: „Deine letzten angeschauten Kinofilm Streams“->XML Sprachfile
3. Stream feld: zeige die letzten 10 Streams an
4. Stream feld: Klick auf den namen eines der letzten 10 Streams startet diesen wieder an position 0 im Stream Tab (dazu muss die Stream URL (http UND HTTPS) gespeichert werden, sodass der Stream direkt wieder ohne neues Suchen aufrufbar ist
5. Download feld: überschrift: „Deine letzten herunter geladenen Kinofilme“->XML Sprachfile
6. Download feld: klick auf einen Eintrag aus den 10 letzten erfolgreich gedownloadeten Filme, prüft den Downloadspeicherort ob dieser Film dort noch vorliegt und wenn ja, wird der Stream Screen Tab mit diesem lokal gespeicherten Video gestartet, wenn nein: klick startet neue suche nach dieser datei.

Erweiterung:

1. Kostenpflichtig:
    a. YouTube Support
        i. XML Configuration: Enable /Disable YouTube Support und Anzahl YouTube Results
        ii. YouTube Suche über API im Proxy anstoßen, als XML Result in die vorhandene XML Ergebnisse einbetten und ans Handy ausliefern:
            1. YouTube API Key als Konfig im Proxy hinterlegen (=einfache Änderung möglich)
        iii. YouTube Video muss streambar und downloadbar sein
        iv. Youtube Videos sollen ein eigenes Result Icon in den Trefferlisten bekommen und in die Ergebnislisten integriert werden (Sie sollen VOR den Random Results des Content Server stehen: „randomresult“
        v. Beispiel:

```
<typ>Kinofilme</typ>
<name>J Edgar German AC3 DVDRiP XViD REPACK SONS</name>

<category>randomresult</category>
<isdownloadable>1</isdownloadable>
```

2. Kostenlos:
    a. Zu jedem Video als Suchergebnis gibt der content server ein neues Flag aus: isdownloadable -> dieses Flag kommt ins Ergebnis XML und zeigt an, ob für einen Film die Download Option aktiviert ist
    b. Wenn isdownloadable == 0 dann blende bei den suchergebnislisten auf dem Handy die Option „Downloade Video“ incl Icon aus (dieses Video ist dann nur streambar)