**CONTINGENCY PLAN**

**SCENARIO 1**

**Risk:** Google didn’t approve the application to take part in the Play Store

**Description:**

Google Play Store acceptance is necessary when the final application is submitted to Play Store console. After the application is submitted, it generally takes a few hours to get an answer about whether approval or not.

Some very recent policy changes, deviations from content regulations or some technical problems may inhibit the approval process.

**Area of Project:** It takes place after ‘after-development testing’ stage as the (one click) ultimate stage in the publishing process. That’s why this one click stage was not mentioned in the WBS diagram.

**Response Strategy:**

After-development testing stage controls should be done with a great caution. Before attempting to release the application, it is needed to be felt sure about latest Play Store policies. It may also be a good strategy to make this publishing operation at a later hour of the day (based on the local time of the biggest target markets) to earn some time in case of emergency or last minute update necessities occur.

**SCENARIO 2**

**Risk:** Application didn’t get the expected attention/ or worse it got negative reviews

**Description:**

After the application is submitted, first 24 hours is especially important for the application to gain more visibility for the following lifecycle of it. After the initial post-releasing period, if the application doesn’t get enough attention or it gets bad attention, the effects can be devastating for the future of it in the market.

**Area of Project:** ‘Budgeting’ (in Project Management) and ‘Specifying marketing requirements’ (in Business Requirements)

**Response Strategy:**

If the application takes no response from users for the beginning, marketing budget shall needed to be utilized more aggressively for the marketing efforts. More varieties of marketing channels can be used and A/B tests can be done in more meaningful manners to get better ideas about the real cause of the situation.

On the other hand, if bad reviews occur, it will be needed to contact these users as early as possible and get their opinions in details. It is also possible to offer these users free upgrade/premium packs to be able to revise their reviews about the following improvements of the application.

**SCENARIO 3**

**Risk:** Dramatic change in the fiscal policy/shutdown of the MBaaS provider.

**Description:**

Application depends heavily on the backend& database support from the MBaaS provider. If the provider increases its unit prices dramatically ,especially in a high growth/heavy usage by users period, the budgeting and self sustainability of the application may suffer or totally collapse. For an even worse scenario, the provider completely can shutdown its service while not providing enough time to developers to update their systems.

**Area of Project**: “Backend Architecture” (in Application Architecture)

**Response Strategy:**

More flexible architecture should be utilized. The code needs to be easily convertible to another provider’s infrastructure in case of a shifting is inevitable. Price changes may also needed to be mirrored to the end users in a non-irritating ways if the application is not sustainable enough by itself anymore. Considering to create own backend service is another option that can be meaningful according to the value/extend of the application.

M.Kilic Arslan

19.12.2018