**Testing Scenarios to RampUp an International Partner**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Message / Service** | **Description** | **List of Data (ChildKey, EmailAddress, SponsorId etc)** | **Expected Results** | **Pass/Fail** | **Actual Results** |
| **Allocation** |  |  |  |  |  |
| *AllocateChild* | Allocate 4 children in:  4 Programs/ Projects  4 Localchurchpartners  4 Communities  2 countries | 1.  2.  3.  4. | IP receives messages for each allocated child |  |  |
| SOAP GetChildInformation 201310 | For each child |  |  |  |  |
| SOAP GetChildCaseStudy 201310 | For each child |  |  |  |  |
| SOAP GetChildImage 201211 | For each child |  |  |  |  |
| SOAP GetProgramImplementor 201309 | For each new project |  |  |  |  |
| SOAP GetCDSPImplementor 201309 | For each new project |  |  |  |  |
| SOAP GetProjectAgeGroup 201309 | For each new project |  |  |  |  |
| SOAP GetLocalChurchPartner 201309 | For each new church partner |  |  |  |  |
| SOAP GetCommunity 201309 | For each new community |  |  |  |  |
| GetCountryInformation 201104 | For each new country |  |  |  |  |
| **Commitments** |  |  |  |  |  |
| *CreateConstituent* | Sent from IP  Must wait for a ConstituentCreated message before creating a commitment |  | ConstituentCreated message received for each message sent |  |  |
| *CreateCommitment* (ChildSponsorship) |  |  | CommitmentCreated message received for each message sent |  |  |
| *CreateCommitment* (ChildCorrespondenceSponsorship) |  |  | CommitmentCreated message received for each message sent |  |  |
| **Depart and Reinstatement Scenarios** |  |  |  |  |  |
| *DepartChild* | This is a sponsored child. This message comes from Compass. This child will be reinstated. | 1. |  |  |  |
| SOAP GetChildExitDetail 201310 | For the departed child |  |  |  |  |
| *AllocateChild* | This message comes from Compass. This is the departed child. The child may have a different child key. | 1. |  |  |  |
| SOAP GetChildInformation 201310 | For each child |  |  |  |  |
| SOAP GetChildCaseStudy 201310 |  |  |  |  |  |
| SOAP GetChildImage 201211 | For each child |  |  |  |  |
| SOAP GetProgramImplementor 201309 | For each new project |  |  |  |  |
| SOAP GetCDSPImplementor 201309 | For each new project |  |  |  |  |
| SOAP GetProjectAgeGroup 201309 | For each new project |  |  |  |  |
| SOAP GetLocalChurchPartner 201309 | For each new church partner |  |  |  |  |
| SOAP GetCommunity 201309 | For each new community |  |  |  |  |
| SOAP GetCountryInformation 201104 | For each new country |  |  |  |  |
| **Marketing Deallocation Scenario** |  |  |  |  |  |
| *DeallocateChild* (not a transfer) | De-allocate unsponsored child. Marketing provides 3 child keys to de-allocate.  This message is a confirmation from Compass that the de-allocation has happened. | 1.  2.  3. | DeallocateChild message is sent from GMC to IP and does not expect a response |  |  |
| **Transfer Scenario** | Update Child message with transfer and new need\_key  entity\_id stays the same |  |  |  |  |
| *DeallocateChild* (transfer) | Deallocate a child that transferred to a different project with a new church partner. This is initiated by Compass. | 1. | IP receives message |  |  |
| *AllocateChild* | Allocate the transferred child.  This test case is used when a child transfers to a different ICP and the IP receives a new child key.  The test is to ensure that the IP knows this is the same child. | 1. | IP receives message |  |  |
| **Update Child Scenarios From Compass** |  |  |  |  |  |
| *UpdateChild* | Message comes from Compass | 1. | IP receives messages. The Update Child message informs the IP that child information has been updated and the service specified in the Event element should be called to get new information |  |  |
| SOAP GetChildInformation 201310 | For each child |  |  |  |  |
| SOAP GetChildCaseStudy 201310 |  |  |  |  |  |
| SOAP GetChildImage 201211 | For each child |  |  |  |  |
| **Update Project Scenarios From Compass** |  |  |  |  |  |
| *UpdateProject* | i.e. the church started having family fun nights. This is sent when a program row is updated in Compass. | 1. |  |  |  |
| SOAP GetCDSPImplementor 201309 | For updated project |  |  |  |  |
| SOAP GetProgramtImplementor 201309 | For updated project |  |  |  |  |
| SOAP GetProjectAgeGroup 201309 | For updated project |  |  |  |  |
| SOAP GetLocalChurchPartner 201309 |  |  |  |  |  |
| **Update Project Suspended Scenarios From Compass** |  |  |  |  |  |
| *UpdateProject* | Suspended project (in comments only).  In comments, the project is suspended. Project is still active. | 1. |  |  |  |
| SOAP GetCDSPImplementor 201309 | For updated project |  |  |  |  |
| **Constituent Email Scenarios** |  |  |  |  |  |
| *CreateEmail (new)* | For each constituent. A ConstituentCreated message must have been received. |  | EmailCreated message received for each message sent out |  |  |
| *UpdateEmail (new email)* | For a constituent that does not have an email. A ConstituentCreated message must have been received. |  | EmailUpdated message received for each message sent out |  |  |
| *CreateEmail (update)* | For an existing email |  | EmailCreated message received for each message sent out |  |  |
| *UpdateEmail (update email)* | Update existing email |  | EmailUpdated message received for each messages sent out |  |  |
| **Constituent Change Scenarios** |  |  |  |  |  |
| *UpsertConstituent* (insert) | This constituent is not in the system. |  | ConstituentUpserted message received for each message sent out |  |  |
| *UpsertConstituent* (update) | Update of existing Constituent |  | ConstituentUpserted message received for each message sent out |  |  |
| **Sponsor Cancels and Then Comes Back for the same child Scenario** |  |  |  |  |  |
| *CancelCommitmen*t | Cancelled commitment |  | CommitmentCanceled message received for each message sent out |  |  |
| *UpsertConstituent* | Wait for ConstituentUpserted response |  | ConstituentCreated message received for each message sent out |  |  |
| *CreateCommitment* (ChildSponsorship) | Commitment created to the same child |  | CommitmentCreated message received for each message sent out |  |  |
| **Gift Scenarios** |  |  |  |  |  |
| *CreateGift* (BirthdayGift) | Test limits of IP system |  | GiftCreated message received for each message sent out |  |  |
| *CreateGift* (FamilyGift) | Test limits of IP system |  | GiftCreated message received for each message sent out |  |  |
| *CreateGift* (GeneralChildGift) | Test limits of IP system |  | GiftCreated message received for each message sent out |  |  |
| *CreateGift* (ProjectGift) | Test limits of IP system |  | GiftCreated message received for each message sent out |  |  |
| *CreateGift* (FinalOrGraduationGift) | Test limits of IP system |  | GiftCreated message received for each message sent out |  |  |
| **Child Image Tests** |  |  |  |  |  |
| GetChildImage (ImageType is Fullshot) |  |  |  |  |  |
| GetChildImage (ImageType is Headshot) |  |  |  |  |  |
| GetChildImage (ImageType is Portrait) |  |  |  |  |  |
| GetChildImage (ImageType is Raw) |  |  | Returns four fields that identify the head. Ignores all other parameters. |  |  |
| **Load Test** |  |  |  |  |  |
| Xx Children Allocated |  |  |  |  |  |
| Xx Commitments Created |  |  |  |  |  |