|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deliverable** | **Details** | **Output** | **Owner** | **Date** | **Comments** |
| Access to Data lake Web Service | * + This is currently under development per Josh (Unicon)   + Earliest availability (code complete) expected to be end of November | Infrastructure to query/extract application data for training | Unicon | Dec 1? | * + Key Requirement for re-training   + (Bob Long; Unicon) |
| Data Extract to CSV | * + Mary G to work on possible extract to csv instead of data lake | CSV data extract | TechCenter (Mary G Unicon) | Dec 1 |  |
| Web Services Extractor | * + Extract data from DL web services to generate source data for training work   + Utilized OpenID authentication to interact with services | Input data (CSV format) for training | Infiniti? | Dec 11? |  |
| Training Script | * + Python based program to initiate training on historical dataset | Training model for prediction service | Infiniti | Dec 6 |  |
| Prediction Service | * + Python based program to generate classification prediction response for a new application | Application classification response (True/False) | Infiniti | Dec 6 | * + We may modify the output to a probabilistic measure instead of a Boolean response |
| API | * + Provide the access methods necessary for CCCApply to interact with the prediction service | Code & Structure to support prediction service | Infiniti | Dec 11 | * + Need to collaborate with CCCApply to ensure interoperability |
| Architecture Review | * + Arc Review Committee to approve Architecture | Architecture Approval | TechCenter | Dec 13th | * + Josh to provide template   + (John Gray or Carlos will know date) |
| Instance Automation  Requirements | * + Provide the recipe to enable an automated build of training instance. | Document with command-level details. | Infiniti | Nov 20 | * + Will require some fine-tuning if new packages are used later. |
| Instance Automation | * + Build out automated lean compute infrastructure for training | Automation Scripts | Infiniti | ? |  |
| Suspend Process - Integration of Verified, Classified Fraud Responses | * + Fraud responses from prediction service go into suspend process.   + Need a mechanism to tag "actual" classification upon inspection.   + Tagging needs to be integrated someplace in the CCCApply architecture. Need to be able to access tags along with historical data through web services interface | CCCApply Changes? | Unicon | Jan 15? | Critical to deploy continuous re-training |
| Suspend Process - Integration of Escaped Fraud Signatures | * + For new fraud detected by colleges (those not detected by deployed model), colleges will need a mechanism to tag applications as fraud.   + Tagging needs to be integrated someplace in the CCCApply architecture. Need to be able to access tags along with historical data through web services interface | CCCApply Changes? | Unicon | Jan 15? | * + Critical to deploy continuous re-training |
| Code Deployment Methodology | * + Identify method/process for code integration and deployment | Defined process? | Infiniti? | TBD |  |
|  |  |  |  |  |  |