**Ad exchange game workshop**

**Tel-Aviv University 201**6

**Design specification**

# Introduction

The purpose of this document is to present and explain the software implementation of the requirement as defined to the AdX game spac.

We will introduce all the packages and classes we used as a part of our design.

# Design Considerations

## Assumptions

## Constraints

## System environment

We used Java 8.

Our Integrated Development Environment was Eclipse at first, but then we changed to IntelliJ IDEA.

We used Git as source control.

# Architecture

## UCS strategy architecture

## Campaign opportunity strategy architecture

## Bid bundle strategy architecture

The bid bundle strategy architecture use five main classes – BidBundleData, BidBundleStartegy, BidBundleFactorCalculator, BidBundleDataBuilder and BidManager.

In order to calculate the BidBundleData parameters we created the BidBundleDataBuilder class that calls routine from BidBundleFactorCalculator class.

In the BidBundleStrategy class we calculate the different strategies based on the game state and using the BidBundleData parameters, and send the bid strategy to the bid manager that sends the bid bundle to the server.

There are two more classes that we use for the bid bundle strategy, the first is CampaignBidBundleHistory which saves information about old bid bundles, and KNNBidBundle which we use to calculate the bid bundle Knn factor.

**Bid package**

**Learn package**

# Algorithms

## Bid bundle strategies algorithms

Pseudo-code of the bid bundle strategies algorithms.

### Stable bid algorithm

### First days bid algorithm

### Last days bid algorithm

### Knn factor algorithm

# Database Schema

## Current game database

For each message our agent receives we save information in a class called GameData.

We keep all the campaign reports delivered to the agent. We also keep the initialization messages (PublisherCatalog( and the Initial Campaign Message and the most recent messages and reports ( CampaignOpportunityMessage, CampaignReport, and

AdNetworkDailyNotification.

Using this information we are calculating all the game strategies.

## History database

As a part of our strategies (in bid bundle auction and campaign opportunity auction) we wanted learn from previous games and by that make more educated bid.

For this reason we have the Learn package which include the LearnStorage class.

The LearnStorage class include the following objects –

* List of campaign bid bundle histories – list of objects from class CampaignBidBundleHistory.
* List of campaign opportunity bid histories – list of objects from class CampaignOpportunityBidHistory.
* Map <campaign, bid> of campaign opportunity bids.
* Map <campaign, cost> of campaign costs.
* Array of sent bid bundles.

We update all the objects throughout the game progress and use them for different strategies.

## 