Process Algebra Extension for UxAS
Project Design Document

Document Version: 1.0
**TABLE OF CONTENTS**

1. Authors  
2. Version History  
3. Approvals  
4. Introduction  
5. General Overview and Design Guidelines  
   i.) Constraints and Standards  
6. Architecture Design  
   i.) Software Architecture  
   ii.) Communication Architecture  
   iii.) Performance
1. **AUTHORS**

This document was prepared by:

Angela Scott  
Paul Coen  
Winston Smith

**Team Email:** reservedwords@saluki.onmicrosoft.com

2. **VERSION HISTORY**

<table>
<thead>
<tr>
<th>Version Completion Date</th>
<th>Version Number</th>
<th>Work Completed</th>
<th>Author or Revisor</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0</td>
<td>Initial draft</td>
<td>Angela Scott, Paul Coen, Winston Smith</td>
</tr>
</tbody>
</table>

3. **APPROVALS**

<table>
<thead>
<tr>
<th>Approval Date</th>
<th>Version Approved</th>
<th>Name and Title of Approver</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0</td>
<td>Bardh Hoxha</td>
</tr>
</tbody>
</table>

**Date:**  
Approved by:  
Approver Signature:  
Mentor Name:  
Mentor Signature:  

4. **INTRODUCTION**

The purpose of this document is to provide a structure which we will build upon to create our code. The goal of this document is to get feedback on our plan so far from the customer to make sure that their needs are satisfied and our goals focused throughout this project.

5. **GENERAL OVERVIEW AND DESIGN GUIDELINES**

i.) Constraints and Standards  
[https://docs.google.com/document/d/1c-kT88X3ibDEkWzGXK6NsT8uK1c0lXB4MiJX5QiH2Gs/edit?usp=sharing](https://docs.google.com/document/d/1c-kT88X3ibDEkWzGXK6NsT8uK1c0lXB4MiJX5QiH2Gs/edit?usp=sharing)

6. **ARCHITECTURE DESIGN**

i.) Software Architecture  
UxAS uses a highly modular system with Services, Utilities, Tasks, and Communications. Our changes took place almost entirely within a few of the Services, AssignmentTreeBranchBoundBase and
PlanBuilderService. Each of these has a .cpp and a .h file; we edited the .h files only to include fields necessary to allow the changes we made to the .cpp files.

ii.) Communication Architecture

UxAS uses messages, analogous to Linux signals, for communication between its various modules. We edited one type of message to include some information we needed to be passed between two Services.

iii.) Performance

Since UxAS uses a greedy, brute-force algorithm, branch and bound, to parse many nodes which represent possible assignments for the vehicles, efficiency per node is a very major concern. Our changes are one-time or single-comparison, and we estimate that we have added less than 0.01 seconds overall to the running of UxAS.