|  |  |
| --- | --- |
|  | **2015** |
|  | Knowledge Based Systems  Possible Phones Helpdesk Bot  Semester 01 Academic year 14/15  Name: Tin Phan, Iulian Bogdan Iancu, Gurjote Khalsa  Student Number: 21246688, 21240924 ,21246685 |
|  |  |

|  |
| --- |
| **[Software conversion documentation]** |
| **Knowledge Based Systems Semester 01 Academic year 14/15** |

Table of Contents

[1 Motivation 2](#_Toc440270592)

[2 Present System description 2](#_Toc440270593)

[3 Intended Changes 3](#_Toc440270594)

[4 Knowledge representation 3](#_Toc440270595)

[5 Expected System functionalities 5](#_Toc440270596)

[6 Test documentation 5](#_Toc440270597)

[7 Evaluation of the adapted system 5](#_Toc440270598)

[8 Github 5](#_Toc440270599)

# Introduction

In this documentation, we will evaluating the project and our motivation

# 1 Motivation

In this section you will describe your motivation to adapt the given QuestionBot 2.x System to create your system working in the domain you want to design.

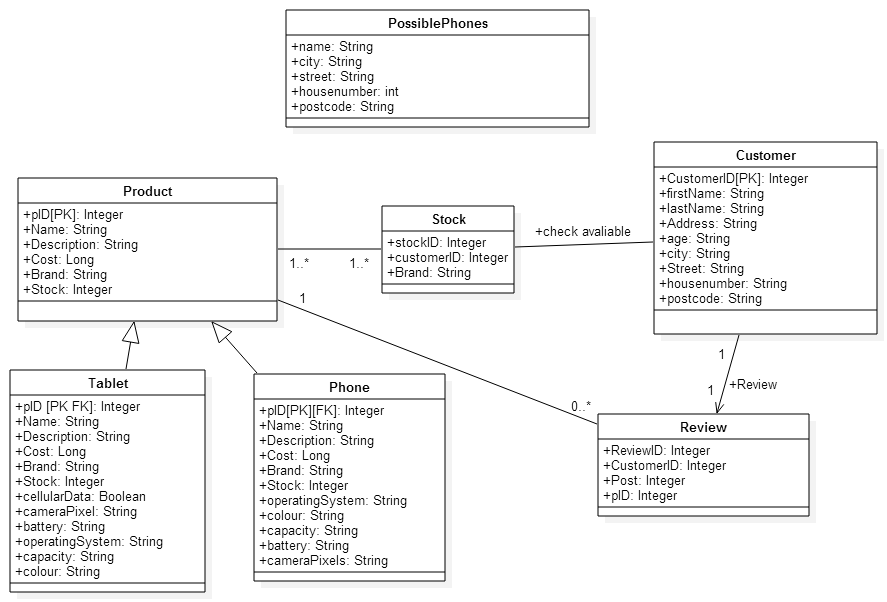
Our intention is for the modern people to understand what they are purchase without the need for human interaction. Allow a SMART machine to assist on the customer purchasing their desired phone.

To be able create a Phone Helpdesk for the possibilities to choose an array of phones. The motive assist for a customer and have their chosen desired new phone. As the helpdesk assist choosing their particular phones and its’ specifications.

# 2 Present System description

In this section you give a brief overview of the functionalities of the current QuestionBot system. This shouldn’t be to elaborate but still be able to give an insight into the inner working of the QuestionBot classes. You might use **UML diagrams** here.

UML Class Diagram



The UML Diagram, consist of all our classes, relationships and attributes.

# 3 Intended Changes

Within this section you give a brief description of the changes you intend to apply to the QuestionBot to adapt it into your new Help desk application. In this section you focus on changes to the program (if any) like added or removed functionalities. For example if your program will be able to answer more (different) questions after the conversion then before you should describe the kind of questions in this section.

We have removed the catalog and lending from the question bot.

We have added two libraries of phones and tablet

We decided to implement the reviews for the products

# 4 Knowledge representation

Here you describe the formalization of the knowledge within the domain you choose to adapt the QuestionBot for. You can do this by providing an **ER- or UML- diagram** or describing the main entities and their relationships within your domain in plain text. Additionally you should provide insight into your knowledge formalization by at least **providing the changed parts of your XML Schema here with remarks on what you changed and why**.

In the XML schema we have added Phones, Tablets, and the review classes. This allows the customer to view the availability of product Phones and Tablets and the product detail for each product.

We have changed the classes to revolve our domain, the Possible Phones Helpdesk Bot; alongside with the attributes, with additional attributes added and value added.

# 5 Expected System functionalities

This section gives a brief overview about the basic functionalities you expect the system to provide after the conversion. For example you can describe these in examples. “The system is expected to be able to provide a sensible answer to a user who asks if he can prolong his lending of a videogame.”

Remember to list the complete Set of the expected functionalities here.

# 6 Test documentation

(To be done after the conversion of the software)

Here you provide **Screenshots**, and or other prove of the program running and artefacts providing information about the outcome of your testing of your conversed program. This kind of artefacts can be tables with the amount of questioned and correctly answered questions or error logs, in short anything that documents the performance and functioning of your converted project.

# 7 Evaluation of the adapted system

(To be done after the conversion of the software)

In this section you evaluate the program by presenting it to at least one test person. You document the feedback the test person gives upon the converted program. You can also detail on changes to your final program which you applied during the conversion and why you changed your conversion plans. “I wanted to implement feature X, as documented in section “3 Intended Changes” but had to abandon these approach because I wasn’t able to provide the necessary knowledge formalization using XML Schema.”

# 8 GitHub

To updated repository. Version Control

Merging branches to allow other collaborators monitor our version control we have used a system specifically equipped for version control, GitHub. GitHub is a software to input project repositories. As you can collaborate with other users on many project as you wish. ‘Commit’ means changes towards the code would need to be updated in the GitHub. Logging each commits each user by ‘push’ to update the current system when changes. ‘Pull’ is also used to grab the new to join the code, but they may be collision of code. GitHub detects the collision of coding; Allowing data integrity to be implemented. Other advantages of GitHub is widely recommended as the emphasis of speed. And portability.

Repository Link: <https://github.com/Axes3/Phones.git>

Log Entries Link: <https://github.com/Axes3/Phones/commits/master>

In this case you are able to see our repository of ‘Phones’. And the full description and details of each commits. In each case of commits, we input the Summary and description. To ensure others understand what have been modified.