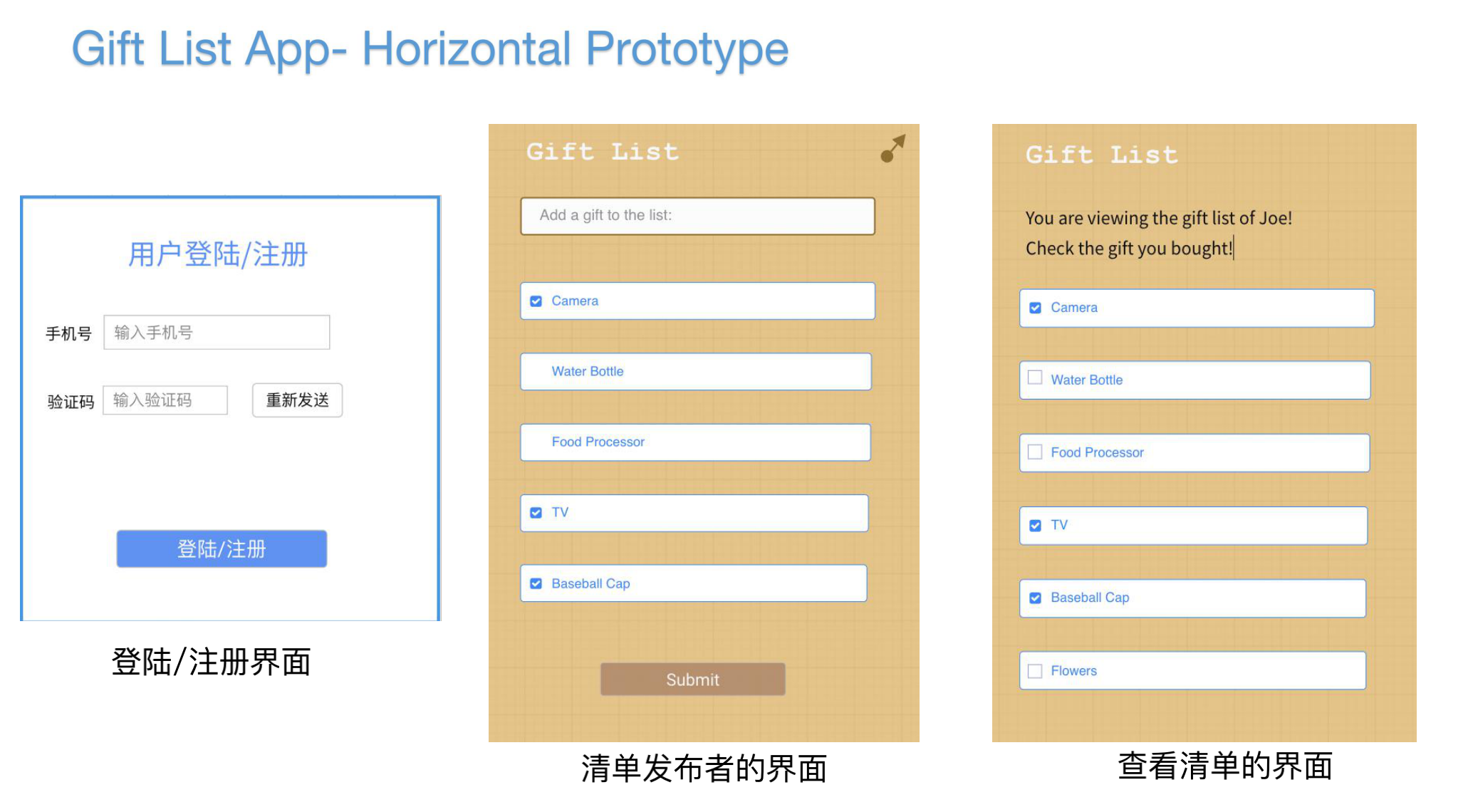
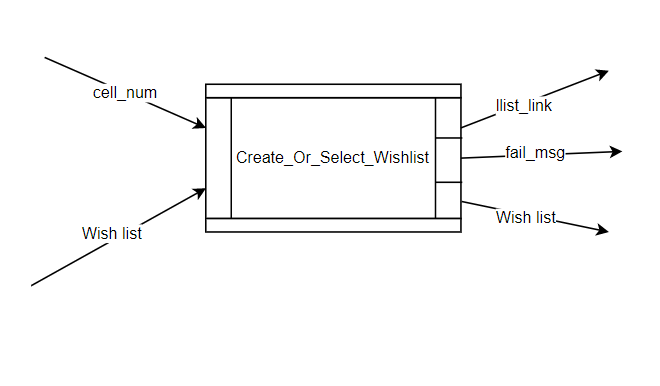
Software Engineering - Experiment 5

Group 3

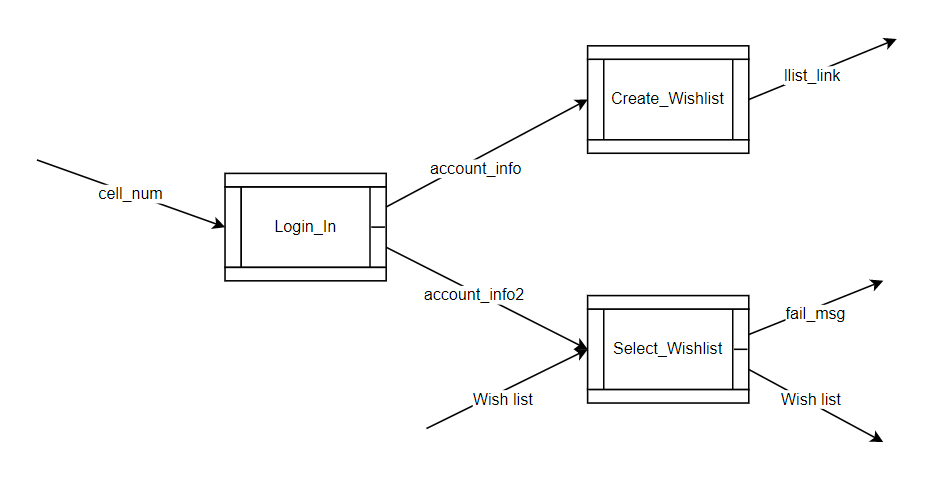
**Horizontal Prototype** is the user interface in the form of screen shots, demonstrating the outer layer of the human interface only, such as windows, menus, and screens. The prototype is used to clarify the scope and requirements of the product.



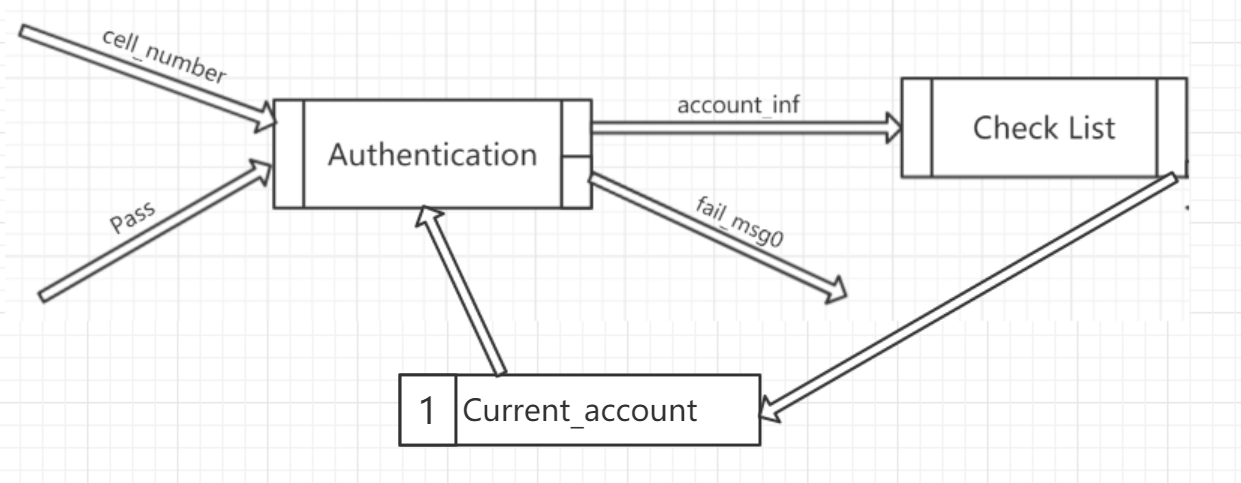
Top:



Level2



Level 3:login



Module *Login*

Type *cell\_number=given pass=given*

Var *current\_account: set of account’s cell\_number(string),password(string),list(set)*

Process *Authentication*

*Fail\_msg: string*

*Account\_info: list*

Ext rd *current\_account*

Post *if cell\_number & password ∊Current\_account*

*Account\_inf = list*

*Else fail\_msg0=”Password does not match the cell number!”*

End\_process;

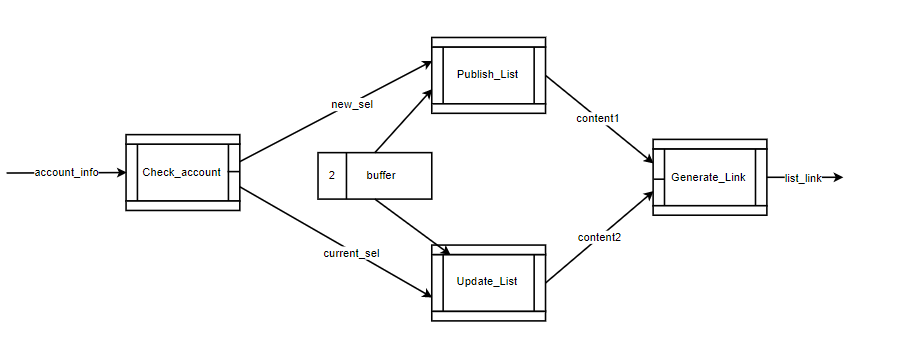
Process *Check\_List*

Post *Current\_account = ~Current\_account*

End\_process;

End\_module;

Level 3: create\_wishlist



Module *Create\_the\_Wishlist*

Type *account\_info = given;*

Var *buffer: List /\*user input\*/*

Process *Check\_Account (account\_info: Account)*

*new\_sel: Boolean | current\_sel: Boolean*

Post *if user press NEW button = True*

*then new\_sel = True*

*else account\_info.list != null*

*then current\_sel = True*

End\_process;

Process *Publish\_List (new\_sel: Boolean)*

*content1: List*

Pre *new\_sel = True*

Post *if new\_sel == True*

*Then content1 = buffer*

End\_process;

Process *Update\_List (current\_sel: Boolean)*

*content2: List*

Pre *current\_sel = True*

Post i*f current\_sel == True*

*Then content2 = buffer*

End\_process;

Process *Generate\_Link(content1:List | content2:List)*

*List\_link: String*

Pre *content1 != null or content2 != null*

Post *if content1 != null*

*then* *list\_link = a new random link*

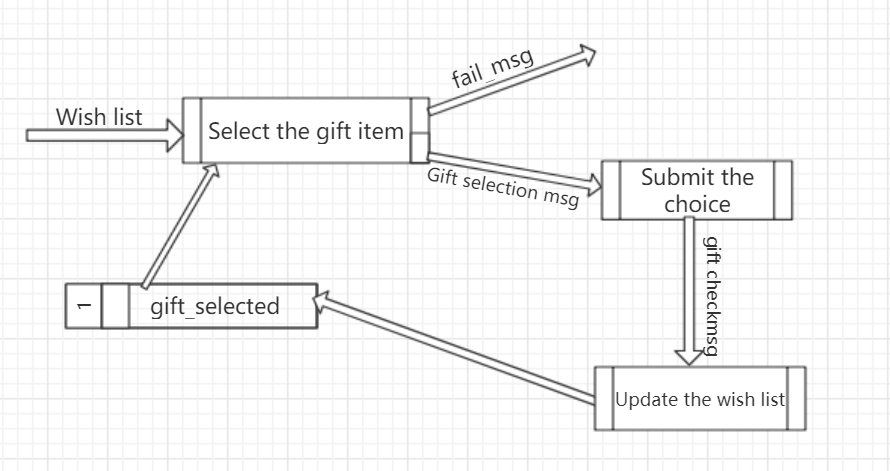
*else*

*then list\_link = orginal list\_link*

End\_process;

End\_module;

Level3: Select\_Wishlist



Module *Select\_the\_Wishlist*

Type *Wish list=given*

Var *gift\_selected: seq of set of gifts*

Process *Select the gift item*

*Fail\_msg: string*

Ext rd *gift\_selected*

Post *if Wish list = true & gift selection msg not in gift\_selected*

*Then Gift selection msg= gift seleciton msg*

*Else fail\_msg=”The wish list you access does not exist!”*

End\_process;

Process *Submit the choice*

Pre *gift selection msg*

Post *if gift selection msg=true*

*Then gift checkmsg=true*

End\_process;

Process *Update the wishlist*

Ext\_rd *gift\_selected*

Post *if gift checkmsg=true*

*Gift\_selected=~gift\_selected - gift checkmsg*

End\_process;

End\_module;

EX5

Defect Checklist

1. Do the data stores cover all the necessary data resources in the software system?

Yes, we have diverse data resources for different processes. Specifically, the cell\_num is the phone number of the user; the buffer is the user input when he wants to modify the current wish lists or to create a new one; the gift\_list corresponds to the current wish list linked with a unique link; more other data resources can be found in the experiment 4.

1. Is there any isolated component in the specification?

No, each component within the system is connected with other components and there is no isolated component in the specification.

1. Is there any inconsistency between processes and their corresponding CDFDs?

No. The architecture of our software system is specified level by level and the system can be decomposed into 4 layers in total. The Top level is the Wishlist System integrating all the functions, then the *Create\_Or\_Select\_Wishlist* function is divided into 3 main modules: *Log\_In, Create\_Wishlist* and *Select\_Wishlist.* On the basis, three modules are further split by different processes. With our examination, there is no inconsistency between processes and their corresponding CDFDs.

4. Is there any inconsistency between the pre-and post-conditions of the connected processes?

As can be seen in the specifications of our three main modules, the pre- and post- conditions of each process are carefully defined and is consistent with the adjacent process. For example in the module *Create\_the\_Wishlist,* the process *Publish\_List* can be activated only if the pre-condition *new\_sel = True* can be satisfied and this is exactly the post-condition of its previous process *Check\_Account.*

1. Do the pre-and post-conditions of each process cover all the possible functional scenarios, especially the ones with exception conditions?

Yes. For example, the item that have been checked already can not be selected by any other user, and the user can only select things in the given wish list. This situation is illustrated in the module *Select\_Wishlist*, if user select items in *gift\_selected,* the system will report a failing message – *“The wish list you accesss does not exist”*.

1. Is there any inconsistency among the modules/CDFDs at different levels?

No.Our top design depict the whole system in a general way, and every sign in it is described with more details in level 2 design. To go on, in the level 2 design, scenarios have been divided into 3 parts, including *Login*, *Create\_Wishlist* and *Select\_Wishlist*. Each part is also provided with diagrams in level 3.

1. Are all the customized types well defined?

Yes. As for the customized types, we use types like string, boolean to restrain and normalize our data. For example, selections are defined as boolean.

1. Are all the system variables well defined?

Yes. For variables, most of them are contained in the database, which often change after the execution of each process or module. We define these database as seq of something or set of something in order to make the information connect closer to matching part.

1. Do all the post-conditions involve both input and output data flows of the corresponding processes?

Yes. For the input, we define *cell\_number*, *pass*, *wishlist* to provide information for processes in order to push the procedure to keep going. And for the output data flows, we consider both the output of next process and the data flows changes in the database.

1. Are all the pre-conditions set as true?

Yes. All the pre-conditions which should be set as true at the very beginning of the process is being checked. And for the triggering conditions, they will be set as true as long as the process goes on in its part.