

HMI: TICKET VENDING MACHINE

Under the guidance of Prof. Dr. Gerd Döben-Henisch and Zeynep Tuncer

-Balasubramaniam Elayaperumal [1143953]
-Pradeep Manjegowda [1144046]
-Sneha Sahu [1142077]

Agenda

The HMI Module

- 1. What is HMI?
- 2. HMI Overview
- 3. Contributors to HMI
- 4. Domains Implementing HMI
- 5. Engineering Process & HMI Phases
- 6. Correct Approach



HMI Subject Matter

Formalization

User Model

Prototyping

Usability Tests

Conclusion

Agenda

The HMI Module

HMI Subject Matter

- 1. Problem Description
- 2. Information at hand

- 3. HMI Expert's Visualization
- 4. User Story



Formalization

User Model

Prototyping

Usability Tests

Conclusion

Agenda

The HMI Module

HMI Subject Matter

Formalization

1. State Graph

2. Formal User Story

3. Automaton

4. Execution Graph



User Model

Prototyping

Usability Tests

Conclusion

Agenda

The HMI Module

HMI Subject Matter

Formalization

User Model

- 1. Umod: Cognitive
- 2. Umod: Rule-Based



Prototyping

Usability Tests

Conclusion

Agenda

The HMI Module

HMI Subject Matter

Formalization

User Model

Prototyping

- 1. Mockup - I
- 2. Mockup - II



Usability Tests

Conclusion

Agenda

The HMI Module

HMI Subject Matter

Formalization

User Model

Prototyping

Usability Tests

- 1. Test Scenario
- 2. Questionnaire
- 3. Heuristic Evaluation
- 4. Thinking Aloud

- 5. Result Comparison
- 6. Method Benefits
- 7. Method Comparison



Conclusion

Agenda

The HMI Module

HMI Subject Matter

Formalization

User Model

Prototyping

Usability Tests

Conclusion

- 1. Deliverables out of HMI
- 2. Industrial Benefits of HMI

- 3. Learnings from the Course



What is HMI?

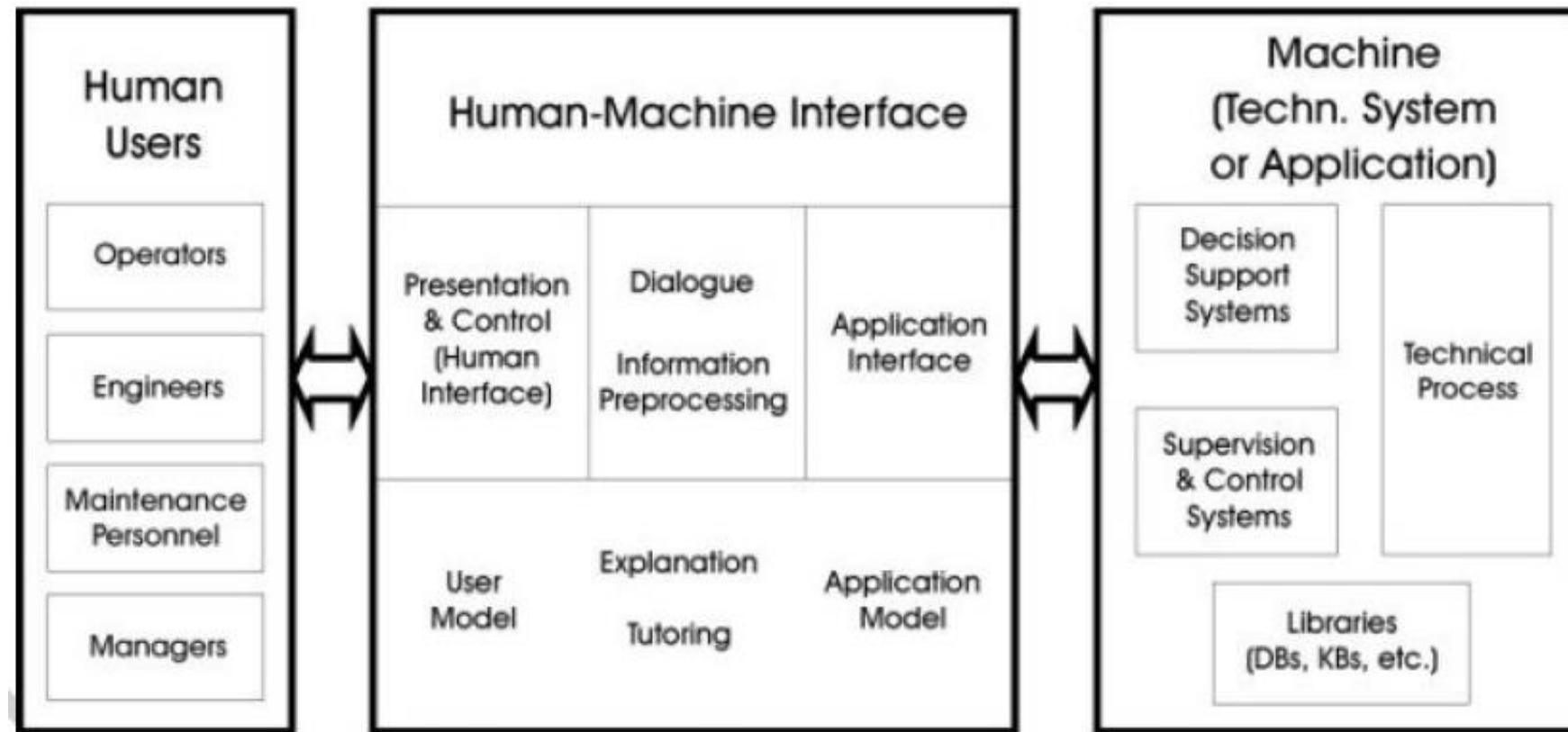
- ❑ HMI: Human Machine Interaction
- ❑ a new discipline of support methodologies for a better human centered layout of any system interfaces. [6]
- ❑ described as the interaction and communication between human users and a machine, a dynamic technical system, via a human-machine interface. [7]



[6] By Gerd Doeßen-Henisch, Zeynep Tuncer & Louwrence Erasmus. As on June 24, 2016. Human Machine Interaction within Systems Engineering A Primer [Draft Version 0.06]

[7] By Gunnar Johannsen. CONTROL SYSTEMS, ROBOTICS, AND AUTOMATION – Vol.XXI – Human Machine Interaction.

HMI Overview [7]



[7] By Gunnar Johannsen. CONTROL SYSTEMS, ROBOTICS, AND AUTOMATION – Vol.XXI – Human Machine Interaction.

Contributors to HMI [7]

HMI requires multiple interdisciplinary views and approaches, the main contributors being –

- **Cognitive Science and Ergonomics**
study of thought, learning, and mental organization, which draws on aspects of psychology, linguistics, philosophy, and computer modelling
- **Automation & System Engineering**
Systems engineering is an interdisciplinary field of engineering that focuses on how to design and manage complex engineering systems over their life cycles
- **Information & Communication Engineering**
Deals with Information Technology, Tele communication, Computers and networks

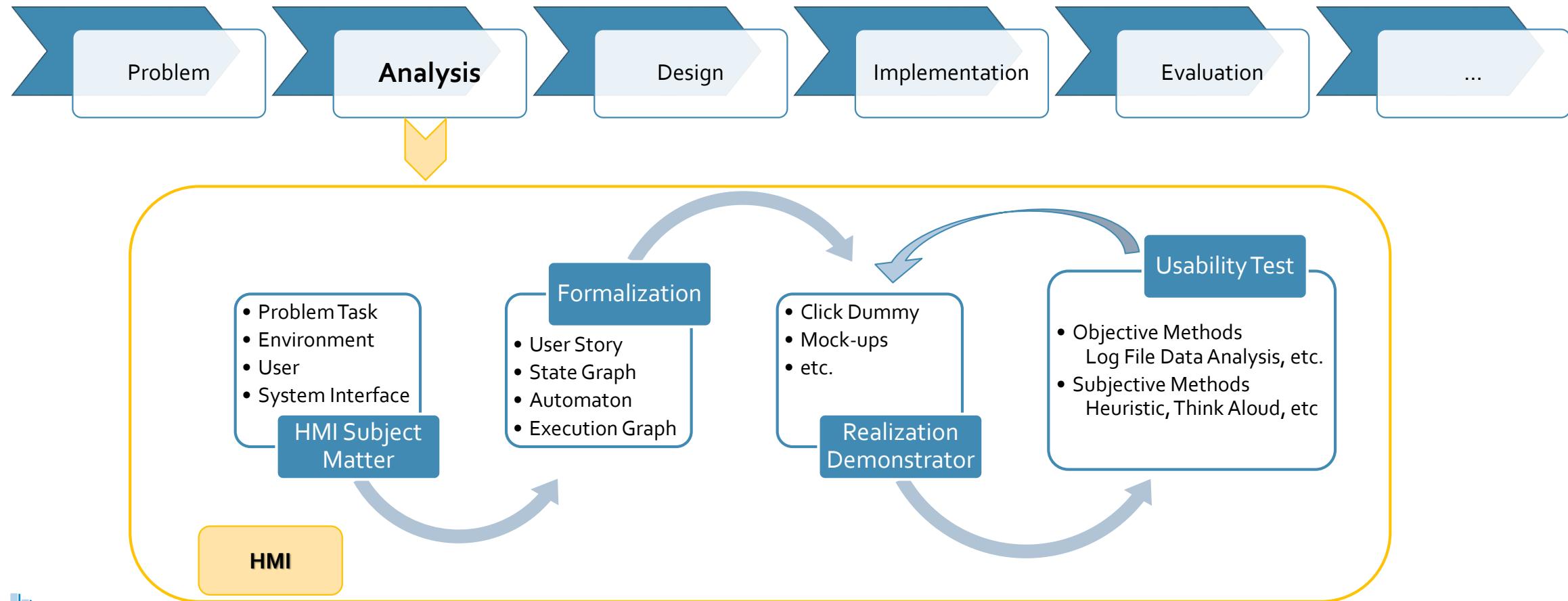
[7] By Gunnar Johannsen. CONTROL SYSTEMS, ROBOTICS, AND AUTOMATION – Vol.XXI – Human Machine Interaction.

Domains Implementing HMI [7]

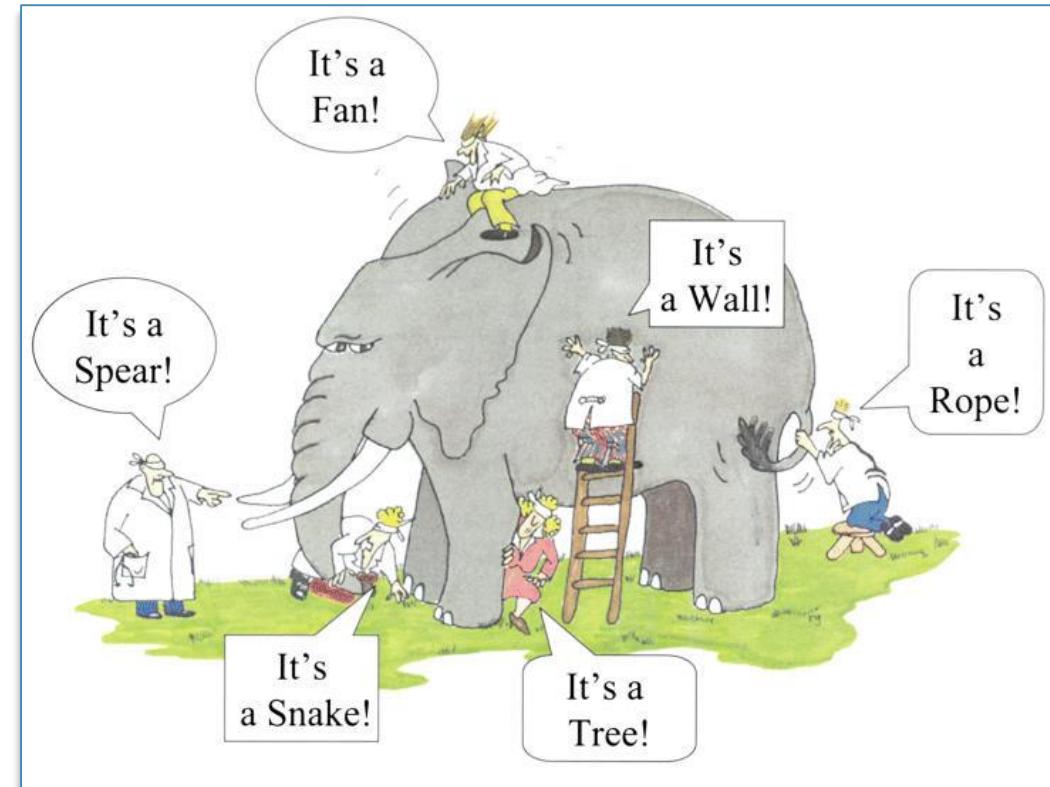
- For many years, aeronautics, astronautics and few car industries widely accepted and used HMI discipline. (Basically Safety Critical Systems)
- At present almost every engineering disciplinary recognized that a good symbiosis between human and machine (HMI) is generally required.
- This wide acceptance of HMI led it to be the part of the Engineering process.

[7] By Gunnar Johannsen. CONTROL SYSTEMS, ROBOTICS, AND AUTOMATION – Vol.XXI – Human Machine Interaction.

Engineering Process & HMI Phases



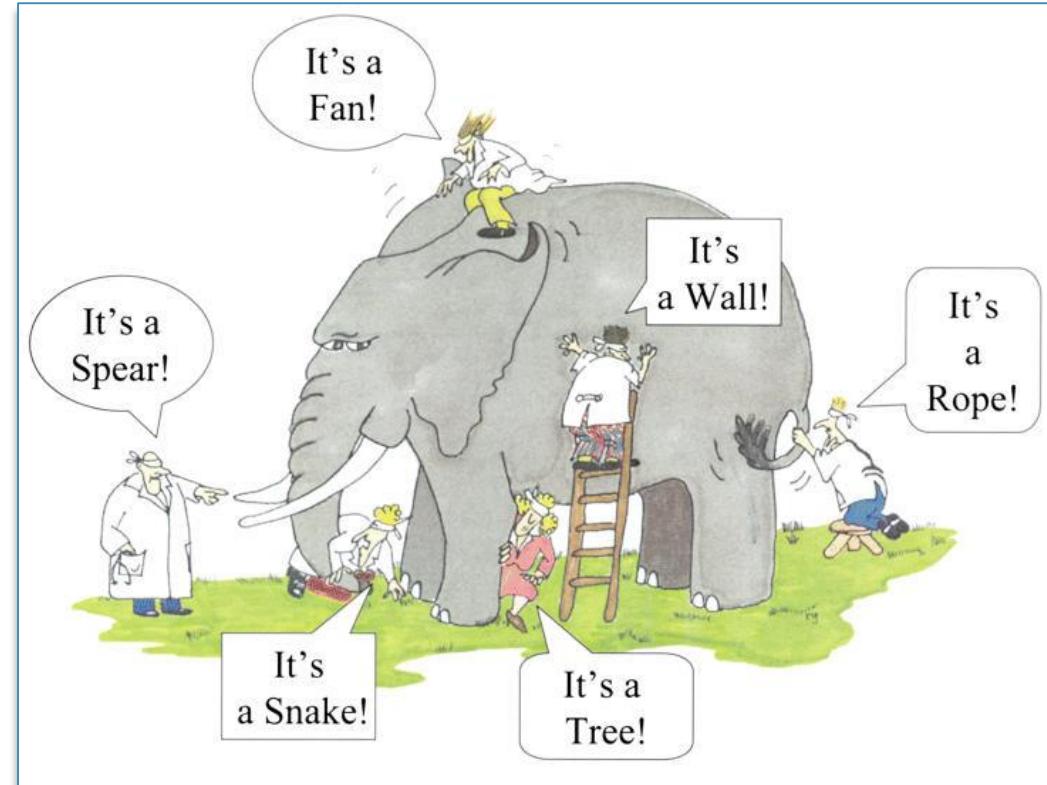
Correct Approach



[10] By Avienaaash Shirali. User Stories: Lack of Big Picture Leads to Blind Man Product. URL: <http://www.agilebuddha.com/agile/user-stories-lack-of-big-picture-leads-to-blind-man-product/>

Correct Approach

- ❑ Lack of bigger picture leads to Blind Man product. [10]
- ❑ too much focus on individual problems may help do an excellent job, but only within THAT specific task.
On the whole the product may still fail.
- ❑ simply knowing that the big picture is an Elephant, the tail will be rightly interpreted as 'tail' and not 'rope'.
- ❑ Importance of User Story

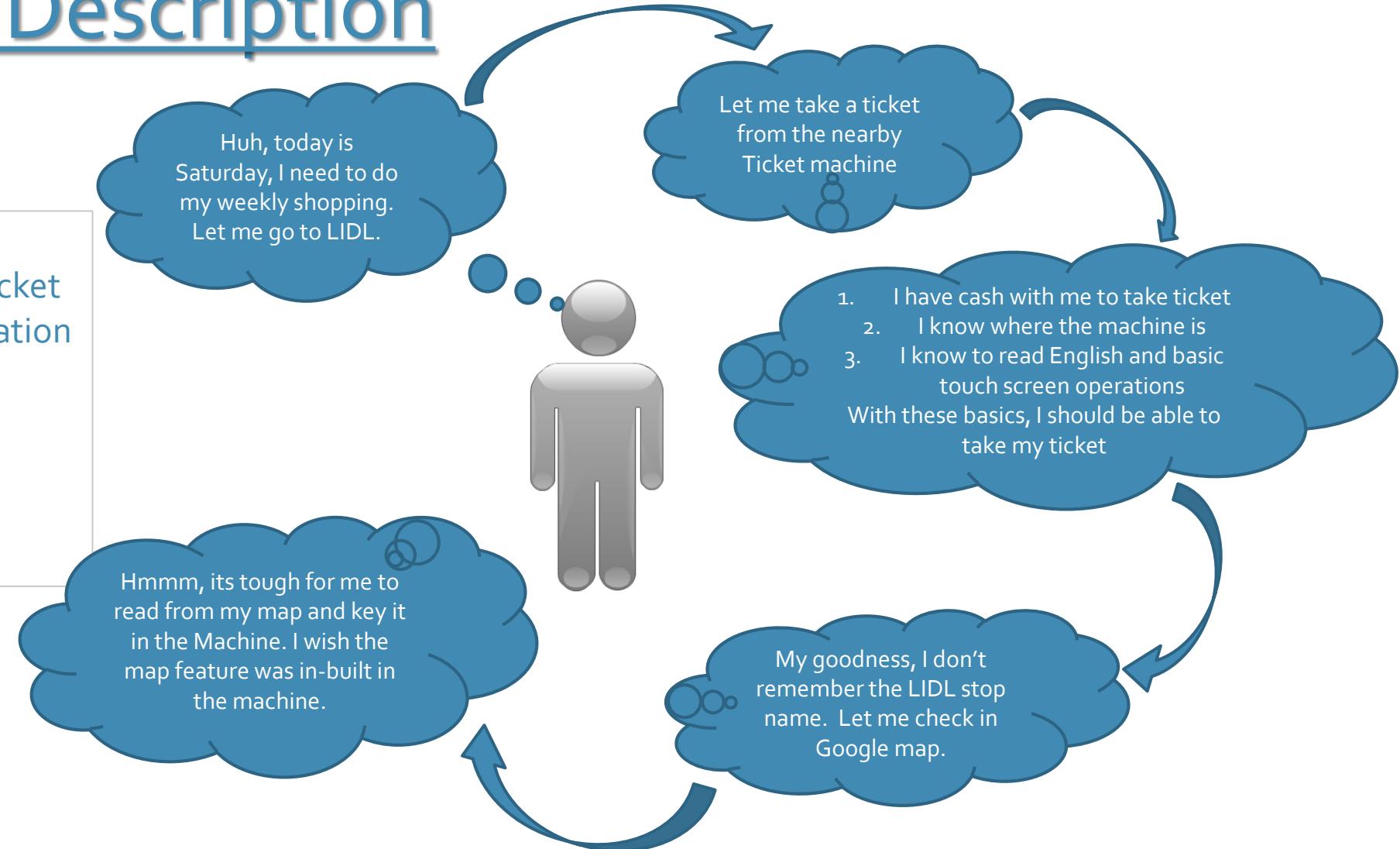


[10] By Avienaaash Shiralige. User Stories: Lack of Big Picture Leads to Blind Man Product. URL: <http://www.agilebuddha.com/agile/user-stories-lack-of-big-picture-leads-to-blind-man-product/>

Problem Description

Problem:

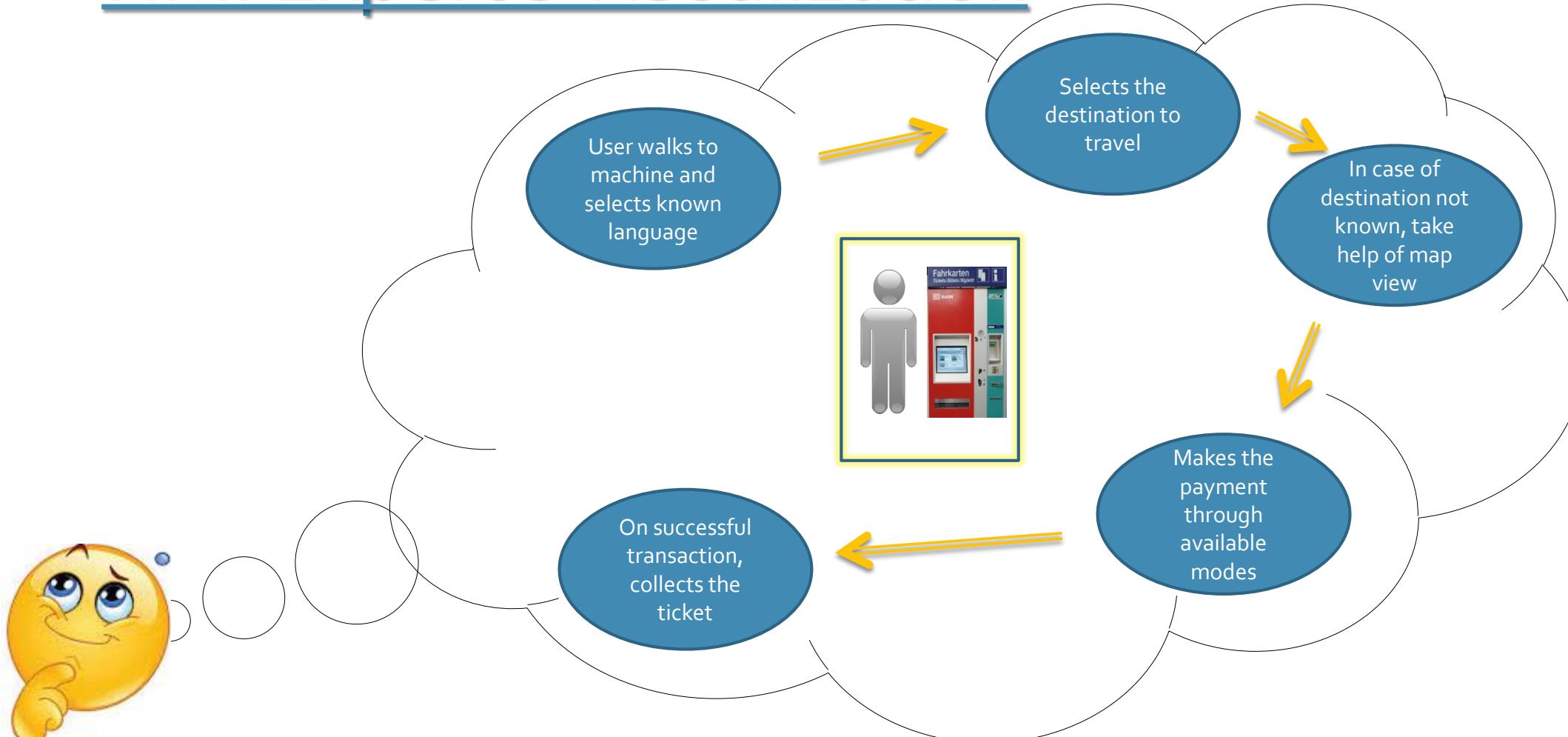
Purchasing a train ticket
when destination station
name is unknown.

Intended solution
Inbuilt map view
functionality

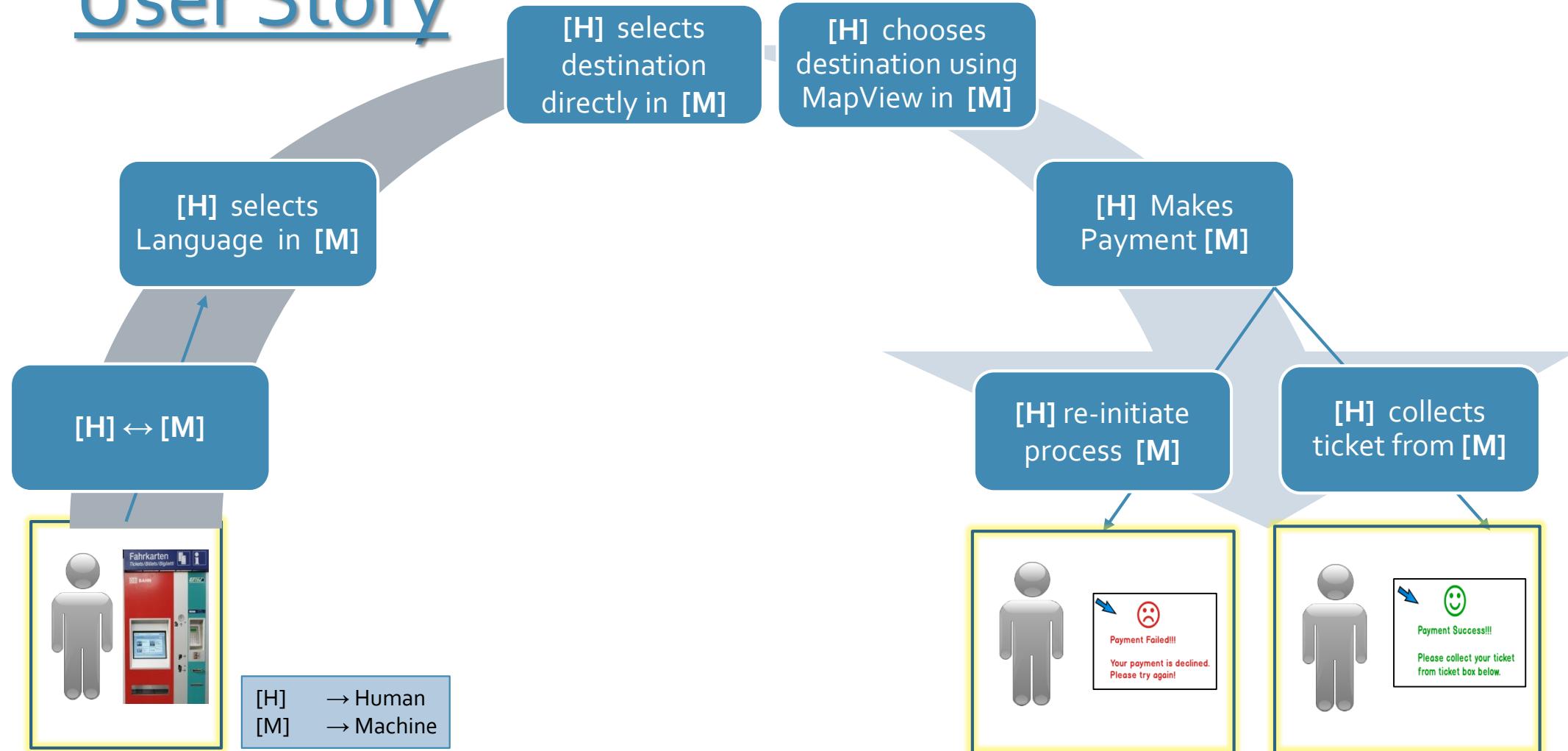
Information at hand

Task	Vending Machine	Human (Actor)	Environment
Get a ticket from source to destination with limited knowledge on destination	Language – English/Deutsch	Knows English	Actor's position is in front of Machine.
	Destination can be selected in 2 ways – Dropdown/MapView	Knows to use touch screen	Machine is in Train Station
	PaymentModes – Cash/Card	Has cash money.	Height of machine is suitable for an average heighited person.
	Prints ticket	Knows how to insert card/cash	Outside temperature is comfortable enough.
	Present at all stations	Knows landmark around destination	

HMI Expert's Visualization



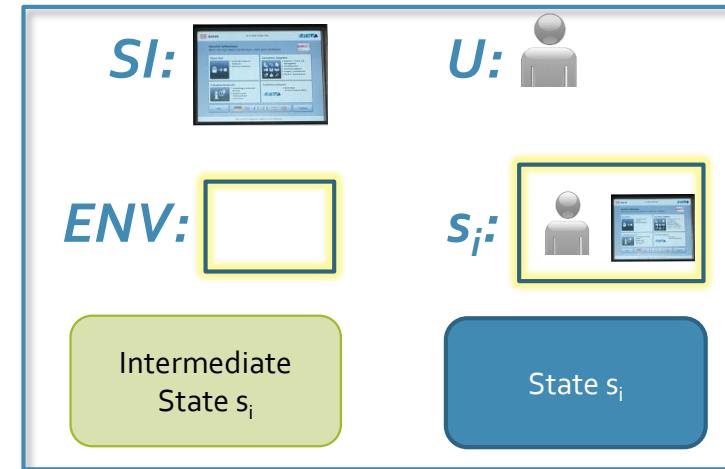
User Story



State Graph (SG)

Each state in the User Story comprises of 3 main sub-states namely – Machine[M]/System Interface[SI], Human[H]/User[U] and Environment[ENV]. [6]

- Each state $s_i = SI \ u \ U \ u \ ENV \ u X$
- Where, SI, U, ENV & X each have properties in the form $p: <<\text{Category}>, <\text{Value}>, [<\text{Unit}>]>$
- To change a state s_i means,
 \exists a finite *set of well defined events* Σ ,
 where *a single event* $\sigma \in \Sigma$ can cause the change of at least 1 property in at least one sub-state.



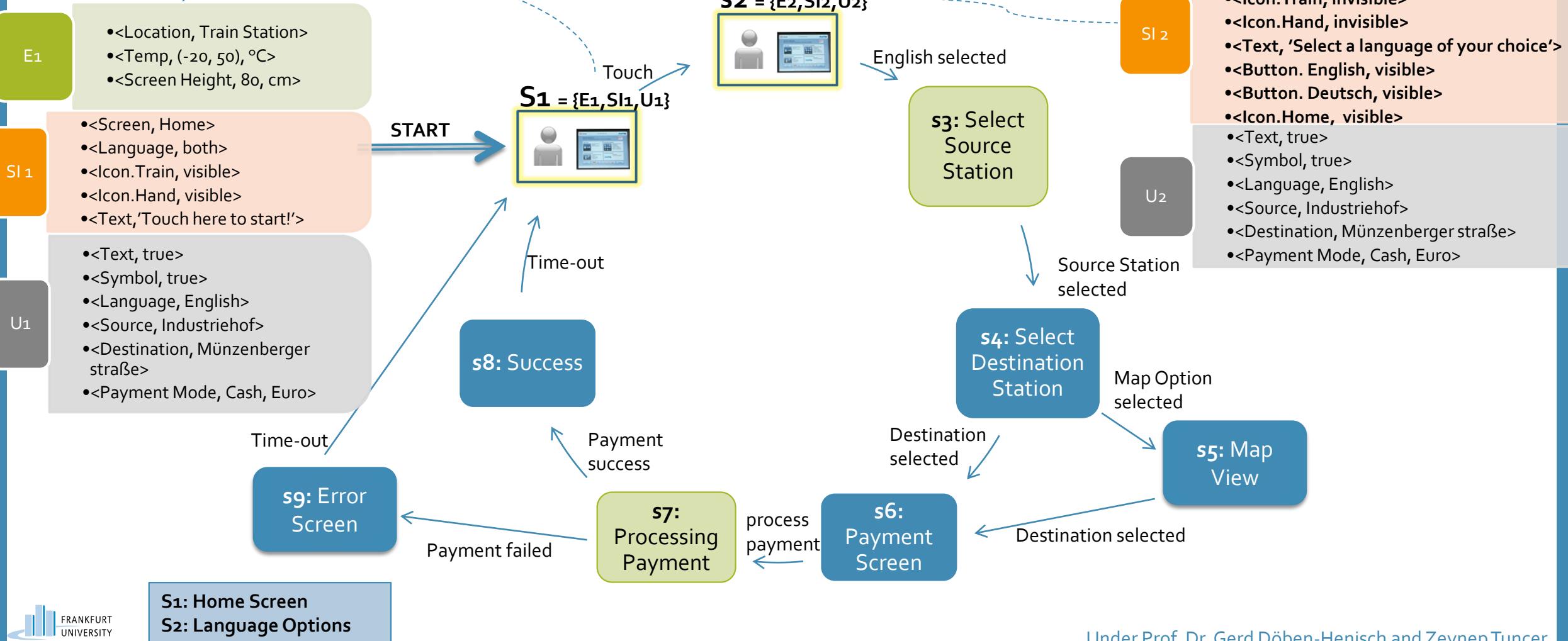
[6] By Gerd Doeblen-Henisch, Zeynep Tuncer & Louwrence Erasmus. As on June 24, 2016. Human Machine Interaction within Systems Engineering A Primer [Draft Version 0.06]

SG: π (set of possible properties)

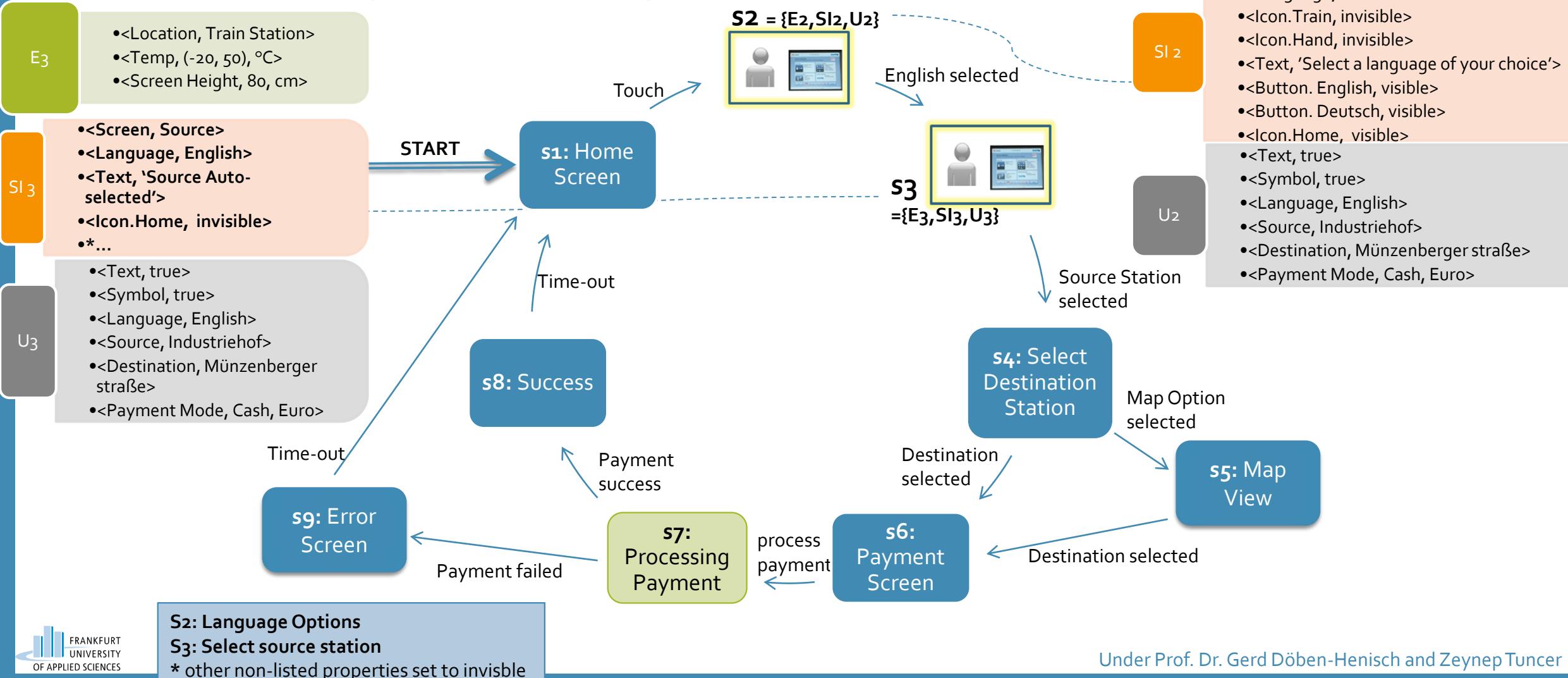
E: Environment	U: User	SI: System Interface	
<Location, Train Station>	<Text, true>	*<Screen, Home Language Source Destination MapView Payment processing success error>	
<Temp, (-20,50), °C>	<Text, false>	*<Icon.Train, visible invisible>	*<Icon.Hand, visible invisible>
<Screen Height, 80, cm>	<Symbol, true>	<Text, 'Touch here to start!'>	*<Icon.Home, visible invisible>
	<Symbol, false>	<Text, 'Select a language of your choice'>	*<Button. English, visible invisible>
	<Language, English>	<Text, 'Source Auto-selected'>	*<Button. Deutsch, visible invisible>
	<Language, Deutsch>	<Text, 'Select your destination place'>	*<Language, English Deutsch both>
	<Language, other>	<Text, 'Click on the destination place'>	*<Button.Map, visible invisible>
	<Source, Industriehof>	<Button.selectedDestination, visible invisible>	*<Button.Destination, visible invisible>
	<Destination, LIDL near FRA-UAS>	<Text, 'Select your payment mode'>	*<Button.Back, visible invisible>
	<Destination, Münzenberger straße>	<Text, 'Want to pay by card? Touch here'>	*<Button.Next, visible invisible>
	<Payment Mode, Cash, Euro>	<Text, 'Want to pay by cash? Touch here'>	*<Button.Cancel, visible invisible>
	<Payment Mode, Card>	<Text, 'Payment failed'>	*<Input.Search, visible invisible>
		<Text, 'Payment success'>	*<Map.googleMap, visible invisible>
		<Label, price, Euros>	<Map, Zoom in/out>
		*<Button.Card, visible invisible>	*<Button.Cash, visible invisible>
		<Text.'Please collect your ticket from the ticket box below'>	

* | : OR notation used to represent multiple values in a category

SG: 2^{π} (state properties)



SG: 2^{π} (state properties)



SG: 2^{π} (state properties)

E3

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>

SI 3

- <Screen, Source>
- <Language, English>
- <Text, 'Source Auto-selected'>
- <Icon.Home, invisible>
- * ...

U3

- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>
- <Destination, Münzenberger straße>
- <Payment Mode, Cash, Euro>

START

s1: Home Screen

Touch

s2: Language Options

English selected

S3
={E3, SI3, U3}

SI 4

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>
- <Screen, Destination>
- <Language, English>
- <Text, 'Select your destination place'>
- <Icon.Home, visible>
- <Button.Map, visible>
- <Button.Destination, visible>
- <Button.Back, visible>
- <Button.Next, visible>
- <Input.Search, visible> *
- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>
- <Destination, LIDL near FRA-UAS | Münzenberger straße>
- <Payment Mode, Cash, Euro>

U4

Source Station selected

S4
={E4, SI4, U4}

Map Option selected

s5: Map View

Destination selected

s6: Payment Screen

process payment

Payment success

s7: Processing Payment

Payment failed

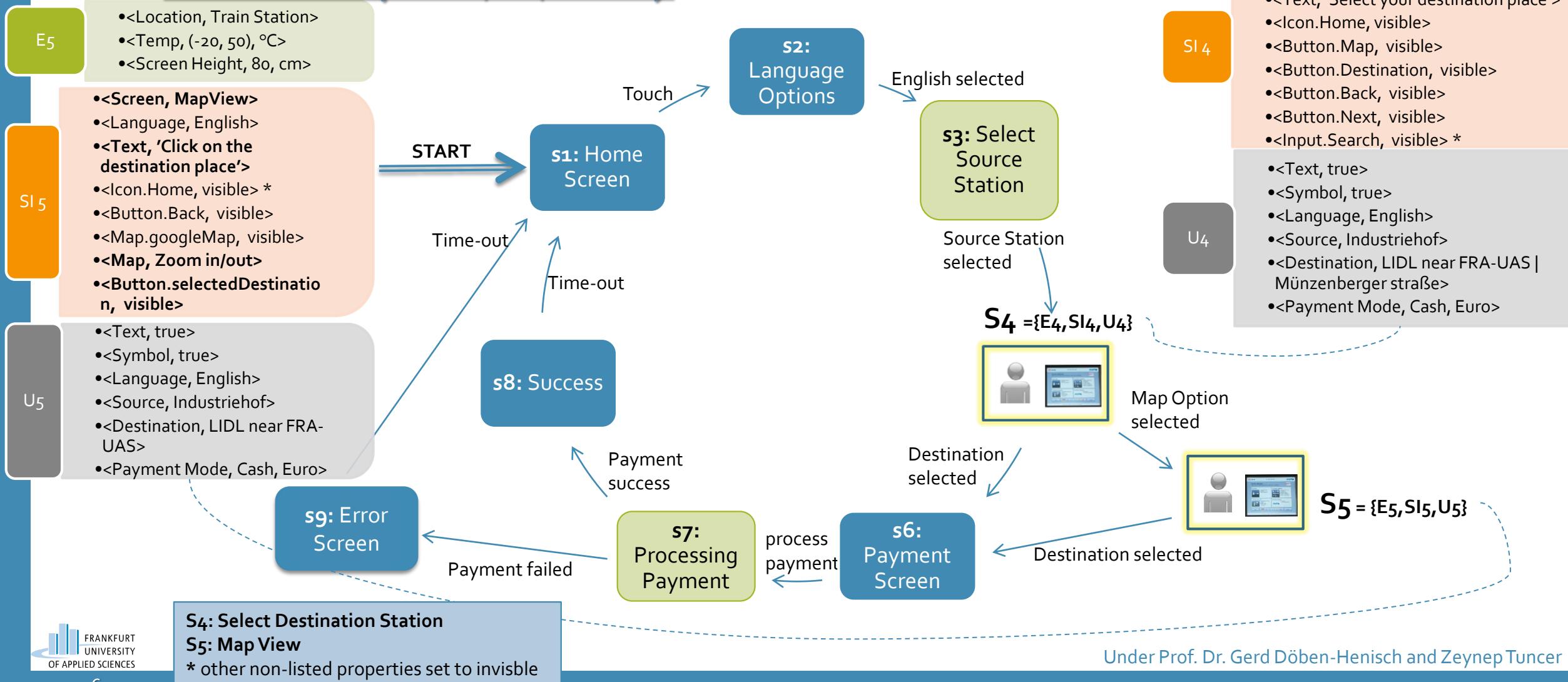
s9: Error Screen

S3: Select Source station

S4: Select Destination Station

* other non-listed properties set to invisible

SG: 2^{π} (state properties)



SG: 2^{π} (state properties)

E5

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>

SI 5

- <Screen, MapView>
- <Language, English>
- <Text, 'Click on the destination place'>
- <Icon.Home, visible> *
- <Button.Back, visible>
- <Map.googleMap, visible>
- <Map, Zoom in/out>
- <Button.selectedDestination, visible>

U5

- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>
- <Destination, LIDL near FRA-UAS>
- <Payment Mode, Cash, Euro>

START

s1: Home Screen

s2: Language Options

s3: Select Source Station

SI 6

U6

s4: Select Destination Station

Time-out

s8: Success

Source Station selected

Destination selected

Map Option selected

s9: Error Screen

Payment failed

s7: Processing Payment

process payment



S6 = {E6, SI6, U6}

S5: Map View
S6: Payment Screen
* other non-listed properties set to invisible

English selected

s3: Select Source Station

U6

s4: Select Destination Station

Map Option selected

S5 = {E5, SI5, U5}

Under Prof. Dr. Gerd Döben-Henisch and Zeynep Tuncer

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>
- <Screen, Payment>
- <Language, English>
- <Text, 'Select your payment mode'>
- <Text, 'Want to pay by card? Touch here'>
- <Text, 'Want to pay by cash? Touch here'>
- <Label, price, Euros>
- <Button.Card, visible>
- <Button.Cash, visible>
- <Icon.Home, visible> *
- <Button.Back, visible>
- <Button.Cancel, visible>
- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>
- <Destination, Münzenberger straße>
- <Payment Mode, Cash, Euro>

E6

SG: 2^{π} (state properties)

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>

- <Screen, Destination>
- <Language, English>
- <Text, 'Select your destination place'>
- <Icon.Home, visible>
- <Button.Map, visible>
- <Button.Destination, visible>
- <Button.Back, visible>
- <Button.Next, visible>
- <Input.Search, visible> *

- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>
- <Destination, Münzenberger straße>
- <Payment Mode, Cash, Euro>

E4

SI 4

U4

s9: Error Screen

Payment failed

s1: Home Screen

Touch

s2: Language Options

s3: Select Source Station

English selected

SI 6

U6

S4 = {E4, SI4, U4}



Map Option selected

s5: Map View

S6 = {E6, SI6, U6}



Destination selected

s7: Processing Payment

process payment

s8: Success

Payment success

S4: Select Destination Station

S6: Payment Screen

* other non-listed properties set to invisible

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>

- <Screen, Payment>
- <Language, English>
- <Text, 'Select your payment mode'>

- <Text, 'Want to pay by card? Touch here'>
- <Text, 'Want to pay by cash? Touch here'>

- <Label, price, Euros>
- <Button.Card, visible>
- <Button.Cash, visible>

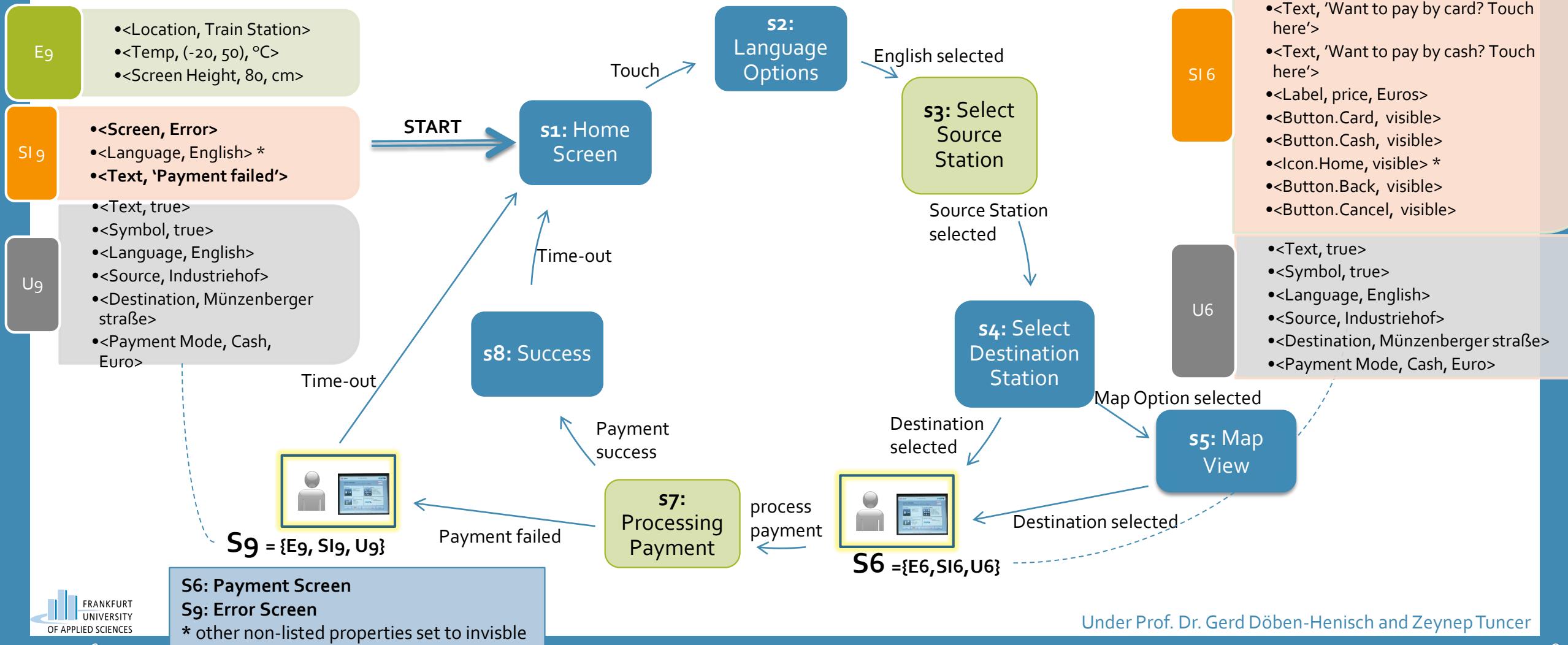
- <Icon.Home, visible> *
- <Button.Back, visible>
- <Button.Cancel, visible>

- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>

- <Destination, Münzenberger straße>
- <Payment Mode, Cash, Euro>

E6

SG: 2^{π} (state properties)



E6

SG: 2^{π} (state properties)

E8

- <Location, Train Station>
- <Temp, (-20, 50), °C>
- <Screen Height, 80, cm>

SI 8

- <Screen, Success>
- <Language, English> *
- <Text, 'Payment success'>
- <Text, 'Please collect your ticket from the ticket box below'>

U8

- <Text, true>
- <Symbol, true>
- <Language, English>
- <Source, Industriehof>
- <Destination, Münzenberger straße>
- <Payment Mode, Cash, Euro>

S6: Payment Screen

S8: Success

* other non-listed properties set to invisible

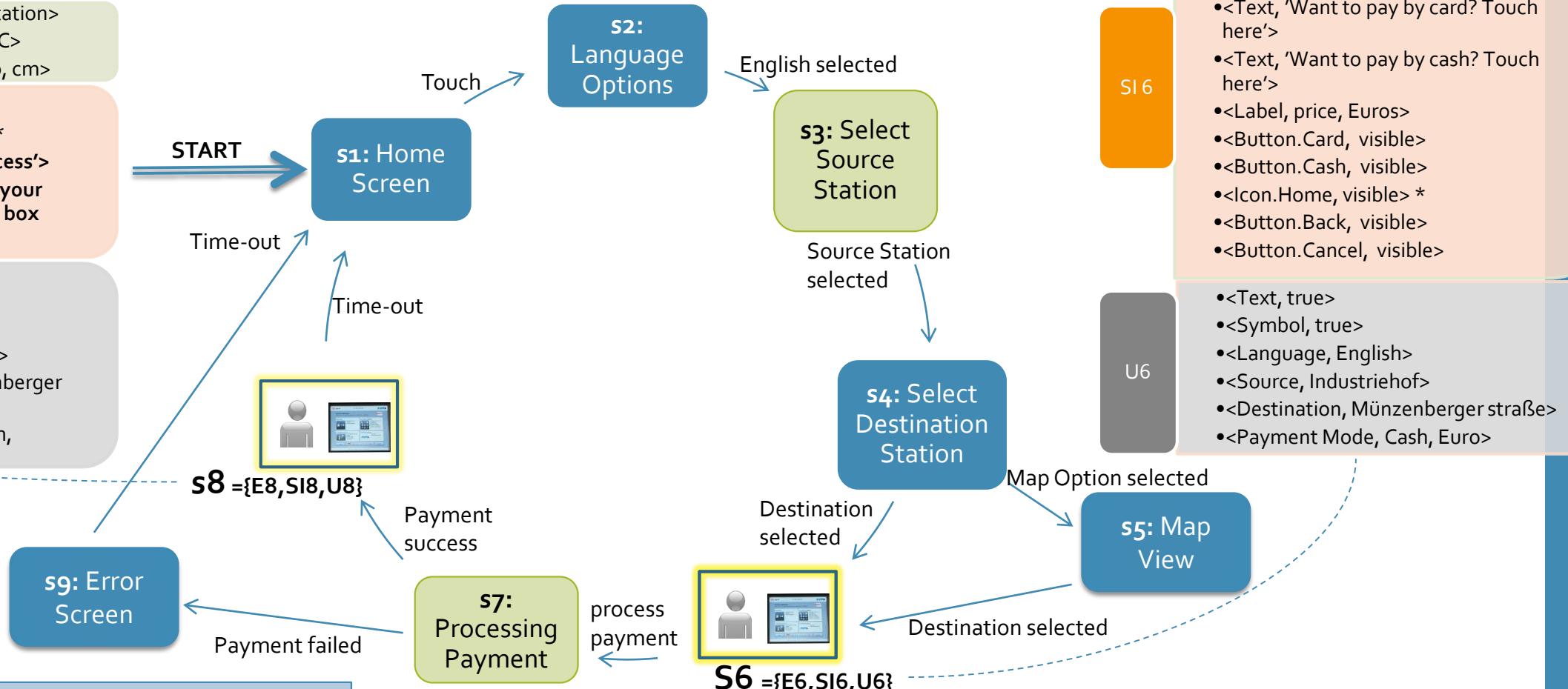
Formalization

User Model

Prototyping

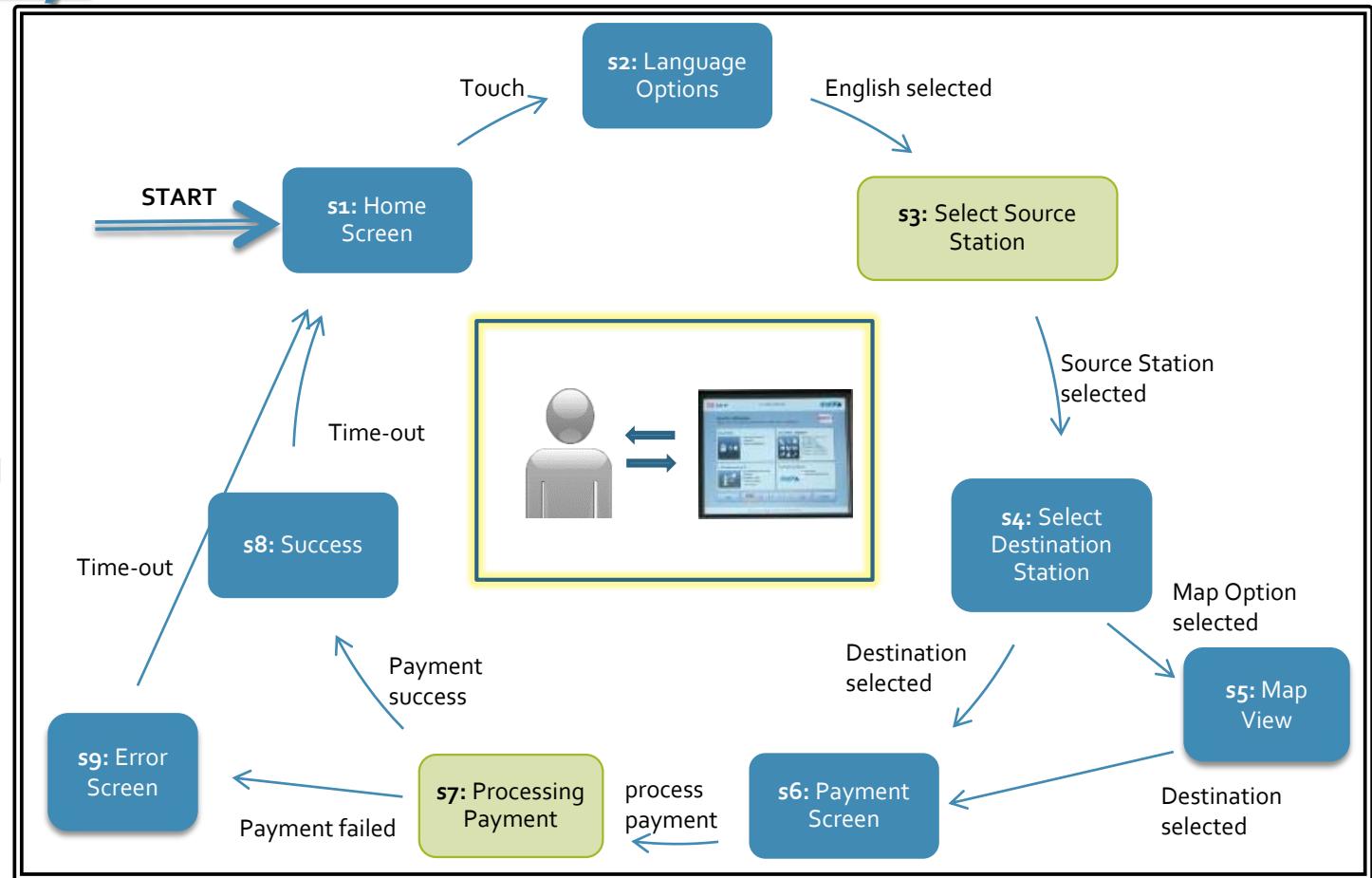
Usability Tests

Conclusion



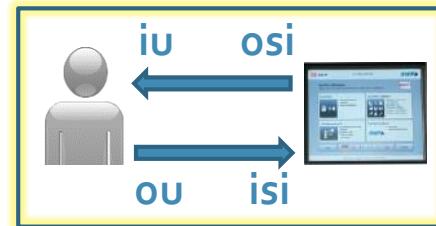
Formal User Story

- Restructured User Story gives more insight into the states.
- Using these state and sub-state properties from SG we will now try to understand the interaction between the sub-states for each State. $[U] \leftrightarrow [SI]$
- The User Story can be formalized using the following notations ^[6] –
 - **Si**: State i, where $i \in \text{Integer}$
 - **osi**: Output from System Interface
 - **iu**: Input to User
 - **ou**: Output from User
 - **isi**: Input to System Interface



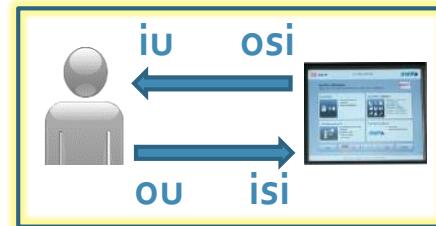
[6] By Gerd Doebe-Henisch, Zeynep Tuncer & Louwrence Erasmus. As on June 24, 2016. Human Machine Interaction within Systems Engineering A Primer [Draft Version 0.06]

Formal User Story



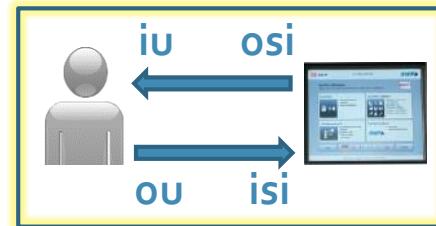
States	Human [H]	System Interface/Machine [M]
S ₁ s ₁ : Home Screen	< <S ₁ ,(ou,touch_icon.train), (osi,'TICKET = pending'),S ₂ > OR <S ₁ ,(ou,∅ touch_any), (osi,'TICKET = pending'),S _{1<td>< <S₁,(iu,'TICKET = pending'), (isi, touch_icon.train),S₂> OR <S₁, (iu,'TICKET = pending'), (isi,∅ touch_any),S₁</td>}	< <S ₁ ,(iu,'TICKET = pending'), (isi, touch_icon.train),S ₂ > OR <S ₁ , (iu,'TICKET = pending'), (isi,∅ touch_any),S ₁
S ₂ s ₂ : Language Options	< <S ₂ ,(ou,touch_button.English touch_button.deutsch), (osi,'TICKET = pending'),S ₃ > OR <S ₂ ,(ou,∅), (osi,'TICKET = pending'),S ₂ > OR <S ₂ ,(ou,touch_icon. home), (osi,'TICKET = pending'),S _{1<td>< <S₂,(iu,'TICKET = pending'), (isi,touch_button.English touch_button.deutsch),S₃> OR <S₂, (iu,'TICKET = pending'),(isi,∅), S₂> OR <S₂, (iu,'TICKET = pending'),(isi,touch_icon. home),S₁</td>}	< <S ₂ ,(iu,'TICKET = pending'), (isi,touch_button.English touch_button.deutsch),S ₃ > OR <S ₂ , (iu,'TICKET = pending'),(isi,∅), S ₂ > OR <S ₂ , (iu,'TICKET = pending'),(isi,touch_icon. home),S ₁
S ₃ s ₃ : Select Source Station	< <S ₃ ,(ou, ∅), (osi,'TICKET = pending'),S ₄ >>	< <S ₃ , (iu,'TICKET = pending'),(isi, ∅), S ₄ >>

Formal User Story



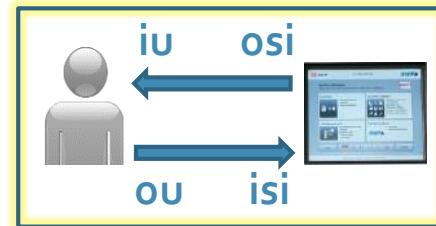
States	Human [H]	System Interface/Machine [M]
S ₄ s ₄ : Select Destination Station	$\langle \langle S_4, (ou, \text{touch_button.destination}), (osi, \text{'TICKET = pending'}), S_6 \rangle \text{ OR}$ $\langle S_4, (ou, \text{touch_button.map}), (osi, \text{'TICKET = pending'}), S_5 \rangle \text{ OR}$ $\langle S_4, (ou, \emptyset \text{touch_search.text}), (osi, \text{'TICKET = pending'}), S_4 \rangle \text{ OR}$ $\langle S_4, (ou, \text{touch_button.back}), (osi, \text{'TICKET = pending'}), S_2 \rangle \text{ OR}$ $\langle S_4, (ou, \text{touch_icon.home}), (osi, \text{'TICKET = pending'}), S_1 \rangle \text{ >}$	$\langle \langle S_4, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.destination}), S_6 \rangle \text{ OR}$ $\langle S_4, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.map}), S_5 \rangle \text{ OR}$ $\langle S_4, (iu, \text{'TICKET = pending'}), (isi, \emptyset \text{touch_search.text}), S_4 \rangle \text{ OR}$ $\langle S_4, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.back}), S_2 \rangle \text{ OR}$ $\langle S_4, (iu, \text{'TICKET = pending'}), (isi, \text{touch_icon.home}), S_1 \rangle \text{ >}$

Formal User Story



States	Human [H]	System Interface/Machine [M]
S5 s5: Map View	$\langle \langle S_5, (ou, \text{button.destination}), (osi, \text{'TICKET = pending'}), S_6 \rangle \text{ OR}$ $\langle S_5, (ou, \emptyset \text{touch_map}), (osi, \text{'TICKET = pending'}), S_5 \rangle \text{ OR}$ $\langle S_5, (ou, \text{touch_button.back}), (osi, \text{'TICKET = pending'}), S_4 \rangle \text{ OR}$ $\langle S_5, (ou, \text{touch_icon.home}), (osi, \text{'TICKET = pending'}), S_1 \rangle \text{ >}$	$\langle \langle S_5, (iu, \text{'TICKET = pending'}), (isi, \text{button.destination}), S_6 \rangle \text{ OR}$ $\langle S_5, (iu, \text{'TICKET = pending'}), (isi, \emptyset \text{touch_map}), S_5 \rangle \text{ OR}$ $\langle S_5, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.back}), S_4 \rangle \text{ OR}$ $\langle S_5, (iu, \text{'TICKET = pending'}), (isi, \text{touch_icon.home}), S_1 \rangle \text{ >}$
S6 s6: Payment Screen	$\langle \langle S_6, (ou, \text{touch_button.card} \text{button.cash}), (osi, \text{'TICKET = pending'}), S_7 \rangle \text{ OR}$ $\langle S_6, (ou, \emptyset), (osi, \text{'TICKET = pending'}), S_6 \rangle \text{ OR}$ $\langle S_6, (ou, \text{touch_button.back}), (osi, \text{'TICKET = pending'}), S_4 \rangle \text{ OR}$ $\langle S_6, (ou, \text{touch_button.cancel} \text{icon.home}), (osi, \text{'TICKET = pending'}), S_1 \rangle \text{ >}$	$\langle \langle S_6, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.card} \text{button.cash}), S_7 \rangle \text{ OR}$ $\langle S_6, (iu, \text{'TICKET = pending'}), (isi, \emptyset), S_6 \rangle \text{ OR}$ $\langle S_6, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.back}), S_4 \rangle \text{ OR}$ $\langle S_6, (iu, \text{'TICKET = pending'}), (isi, \text{touch_button.cancel} \text{icon.home}), S_1 \rangle \text{ >}$

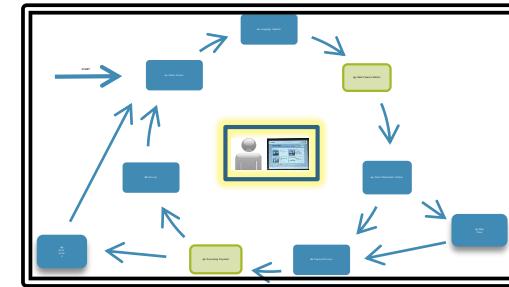
Formal User Story



States	Human [H]	System Interface/Machine [M]
S7 s7: Processing Payment	< <S7, (ou, Ø), (osi, 'TICKET = pending'), S8/S9 >>	< <S7, (iu, 'TICKET = pending'), (isi, Ø), S8/S9 >>
S8 s8: Success	< <S8, (ou, Ø), (osi, 'TICKET = Success'), S1 >	< <S8, (iu, 'TICKET = Success'), (isi, Ø), S1 >
S9 s9: Error Screen	< S9, (ou, Ø), (osi, 'TICKET = Failed'), S1 > >	< S9, (iu, 'TICKET = Failed'), (isi, Ø), S1 > >

Automaton α ^[6]

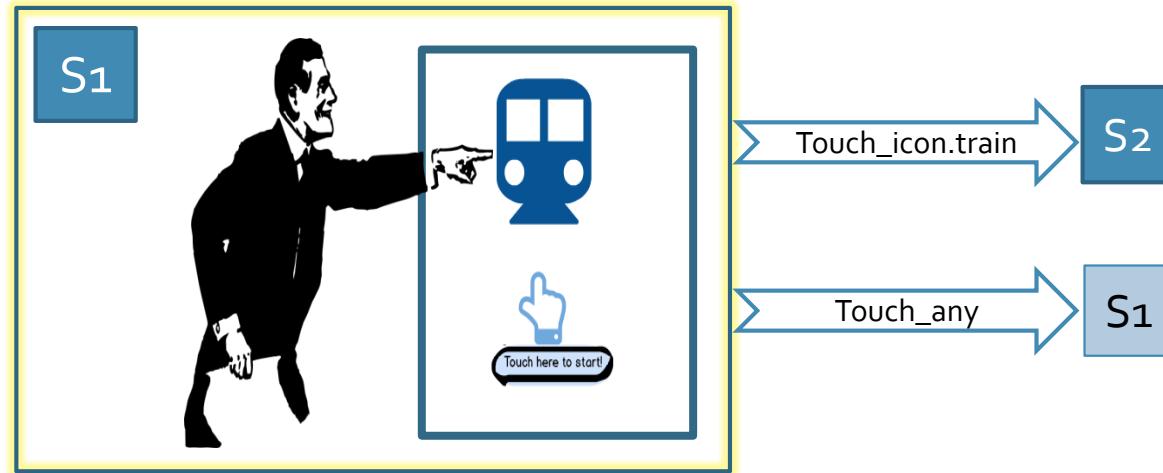
$$\alpha = \langle Q, I, F, \Sigma, \Delta, \pi, \lambda \rangle$$



Set	Notations	TVM Representation
States	Q	$\{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, s_9\}$
Initial States	I	$I \subset Q$ $\{S_1\}$
Final States	F	$F \subset Q$ $\{S_1\}$
Events	Σ	$\{\text{touch_icon.train}, \text{touch_any}, \text{touch_icon.home}, \text{touch_button.back}, \text{touch_button.cancel}, \text{touch_button.Deutsch}, \text{touch_button.English}, \text{touch_button.map}, \text{touch_button.<destination>}, \text{touch_search.text}, \text{touch_map.select}, \text{touch_button.card}, \text{touch_button.cash}, \text{ticket_collect}, \text{payment_success}, \text{payment_failed}, \text{time-out}\}$
Transfer Function	Δ	$Q \times \Sigma^* \times Q$ <Follows...>
Properties	π	(already discussed with state graphs)
Mapped properties	λ	$Q \rightarrow 2^\pi$ <Follows...>

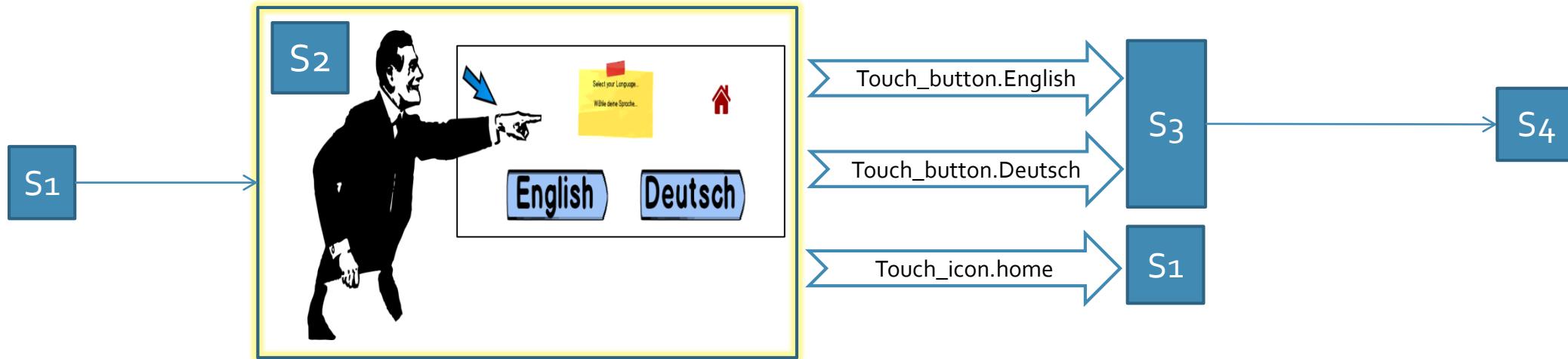
[6] By Gerd Doeßen-Henisch, Zeynep Tuncer & Louwrence Erasmus. As on June 24, 2016. Human Machine Interaction within Systems Engineering A Primer [Draft Version 0.06]

Automaton α



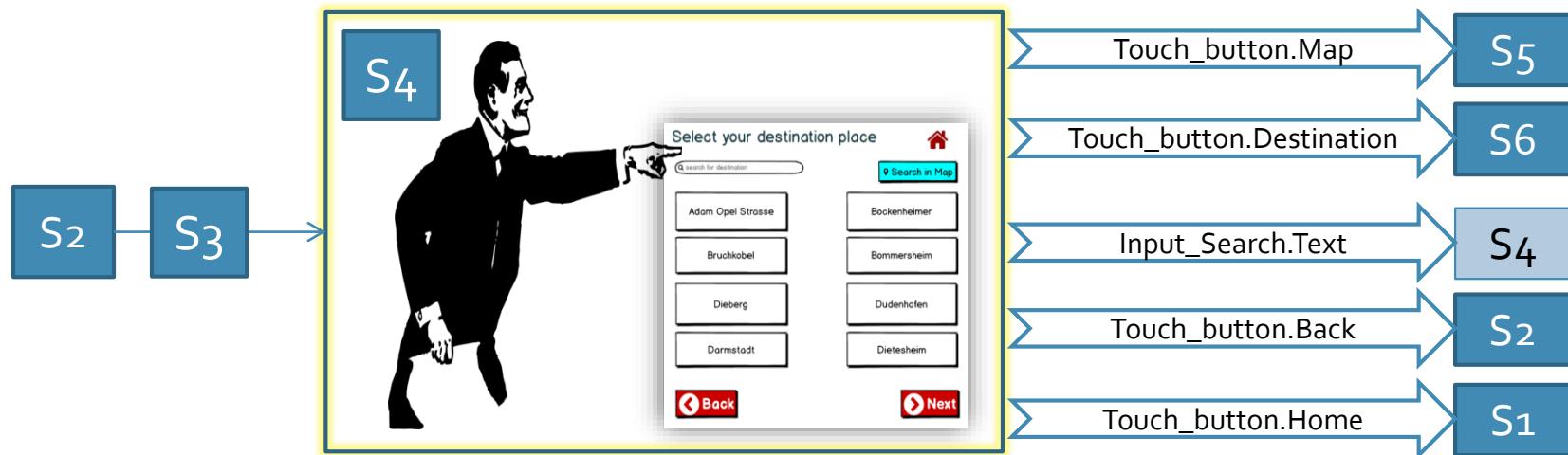
$\lambda: Q \rightarrow 2^\pi$ for $S_1 = \{E_1, S_{l1}, U_1\}$			Transfer Function (Δ)
S_{l1}	U_1	E_1	
<ul style="list-style-type: none"> <Screen, Home> <Icon.Train, visible> <Icon.Hand, visible> <Text,'Touch here to start!'> 	<ul style="list-style-type: none"> <Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, Münzenberger straße> <Payment Mode, Cash, Euro> 	<ul style="list-style-type: none"> <Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm> 	<ul style="list-style-type: none"> <S_1,(touch_icon.train),S_2> <S_1,(touch_any),S_1>

Automaton α



$\lambda: Q \rightarrow 2^\pi$ for $S_2 = \{E_2, S_{I2}, U_2\}$			Transfer Function (Δ)
S_{I2}	U_2	E_2	
<ul style="list-style-type: none"> <Screen, Language> <Icon.Train, invisible> <Icon.Hand, invisible> <Text, 'Select a language of your choice'> <Button. English, visible> <Button. Deutsch, visible> <Icon.Home, visible> 	<ul style="list-style-type: none"> <Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, Münzenberger straße> <Payment Mode, Cash, Euro> 	<ul style="list-style-type: none"> <Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm> 	<ul style="list-style-type: none"> <S_2,(Touch_button.English),S_3> <S_2,(Touch_button.Deutsch),S_3> <S_2,(Touch_icon.home),S_1>

Automaton α



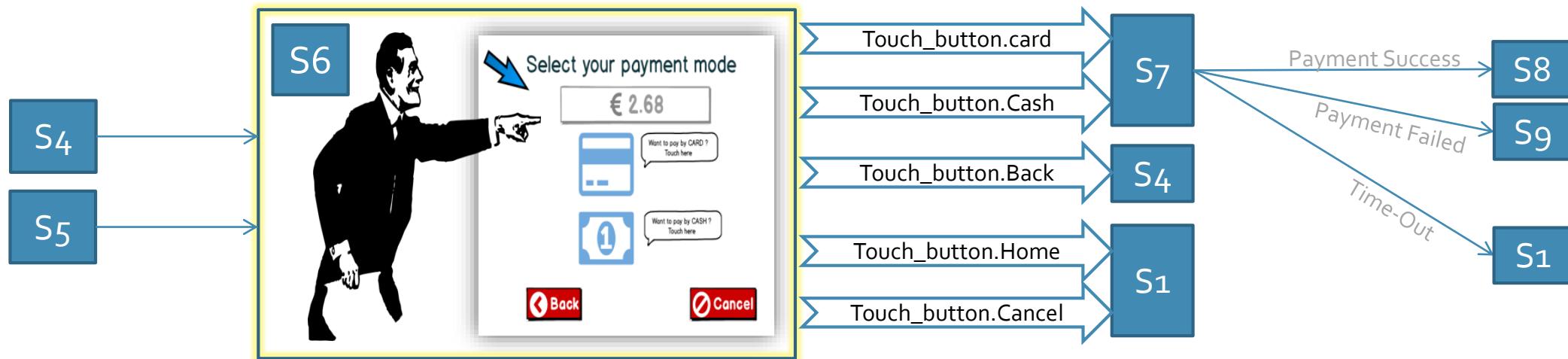
$\lambda: Q \rightarrow 2^\pi$ for $S_4 = \{E_4, SI_4, U_4\}$			Transfer Function (Δ)
SI_4	U_4	E_4	
<ul style="list-style-type: none"> <Screen, Destination> <Language, English> <Text, 'Select your destination place'> <Icon.Home, visible> <Button.Map, visible> <Button.Destination, visible> <Button.Back, visible> <Button.Next, visible> <Input.Search, visible> 	<ul style="list-style-type: none"> <Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, LIDL near FRA-UAS Münzenberger straße> <Payment Mode, Cash, Euro> 	<ul style="list-style-type: none"> <Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm> 	<ul style="list-style-type: none"> <S_4, (Touch_button.Map), S_5> <S_4, (Touch_button.Destination), S_6> <S_4, (input_search.Text), S_4> <S_4, (Touch_icon.back), S_2> <S_4, (Touch_icon.home), S_1>

Automaton α



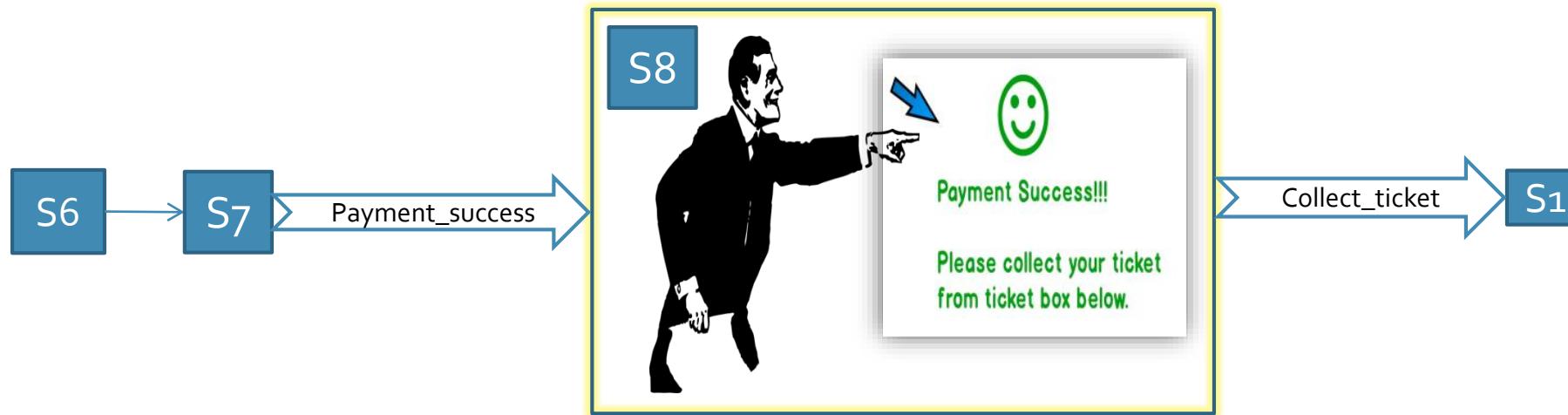
$\lambda: Q \rightarrow 2^\pi$ for $S_5 = \{E_5, SI_5, U_5\}$			Transfer Function (Δ)
SI_5	U_5	E_5	
<ul style="list-style-type: none"> <Screen, MapView> <Language, English> <Text, 'Click on the destination place'> <Icon.Home, visible> * <Button.Back, visible> <Map.googleMap, visible> <Map, Zoom in/out> <Button.selectedDestination, visible> 	<ul style="list-style-type: none"> <Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, LIDL near FRA-UAS> <Payment Mode, Cash, Euro> 	<ul style="list-style-type: none"> <Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm> 	<ul style="list-style-type: none"> <S_5, (Touch_button.Destination), S_6> <S_5, (Map.Select), S_5> <S_5, (Touch_icon.back), S_4> <S_5, (Touch_icon.home), S_1>

Automaton α



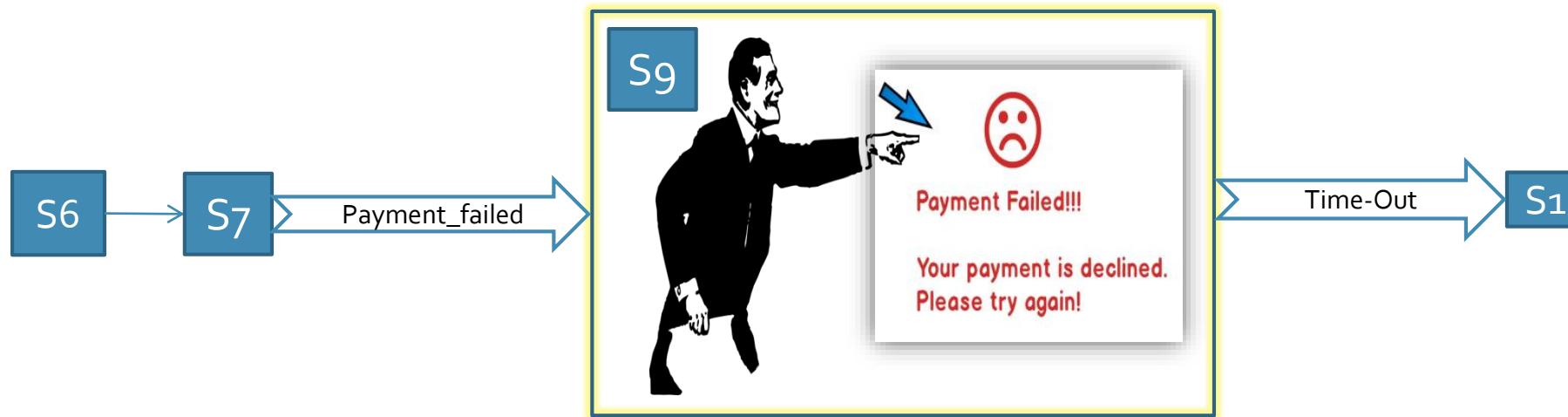
$\lambda: Q \rightarrow 2^\pi$ for $S_6 = \{E_6, SI_6, U_6\}$			Transfer Function (Δ)
SI ₆	U ₆	E ₆	
<ul style="list-style-type: none"> <Screen, Payment> <Language, English> <Text, 'Select your payment mode'> <Text, 'Want to pay by card? Touch here'> <Text, 'Want to pay by cash? Touch here'> <Label, price, Euros> <Button.Card, visible> <Button.Cash, visible> <Icon.Home, visible> <Button.Back, visible> <Button.Cancel, visible> 	<ul style="list-style-type: none"> <Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, Münzenberger straße> <Payment Mode, Cash, Euro> 	<ul style="list-style-type: none"> <Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm> 	<ul style="list-style-type: none"> <$S_6, (\text{touch_button.card})$, S_7> <$S_6, (\text{Touch_button.Cash})$, S_7> <$S_6, (\text{Touch_icon.back})$, S_4> <$S_6, (\text{Touch_icon.home})$, S_1> <$S_6, (\text{Touch_icon.cancel})$, S_1>

Automaton α



$\lambda: Q \rightarrow 2^\pi$ for $S8 = \{E8, SI8, U8\}$			Transfer Function (Δ)
SI8	U8	E8	
<Screen, Success> <Language, English> <Text, 'Payment success'> <Text, 'Please collect your ticket from the ticket box below'>	<Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, Münzenberger straße> <Payment Mode, Cash, Euro>	<Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm>	<S8,(payment_success),S1>

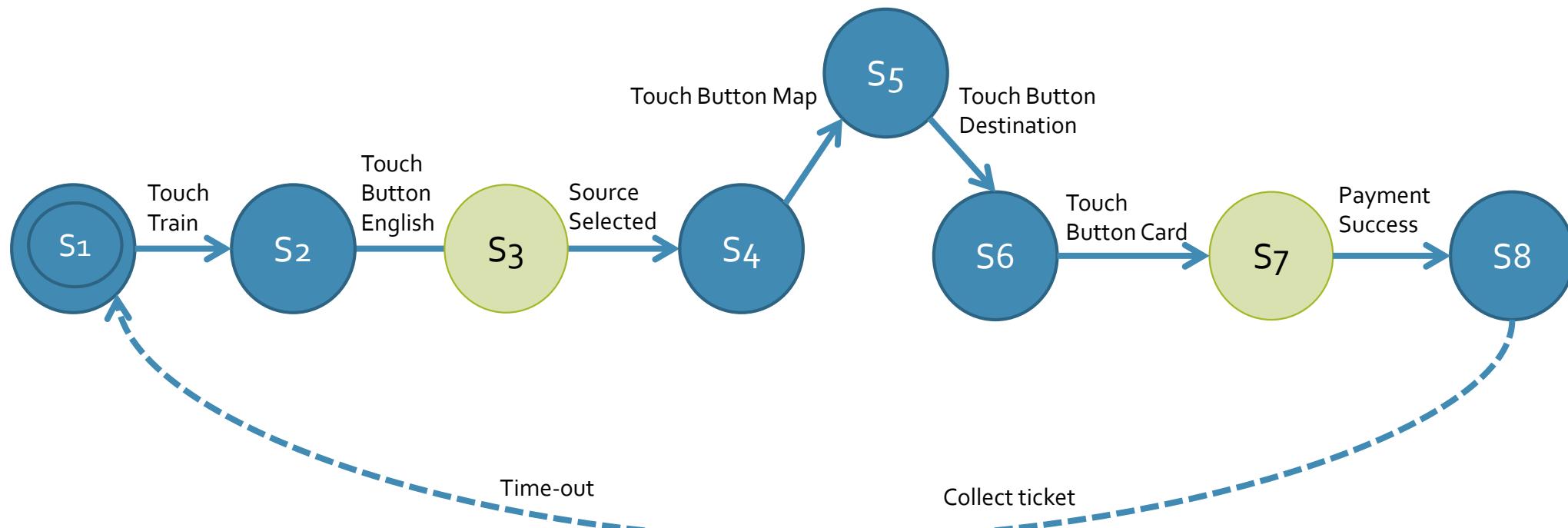
Automaton α



$\lambda: Q \rightarrow 2^\pi$ for $S_9 = \{E_9, S_{19}, U_9\}$			Transfer Function (Δ)
S_{19}	U_9	E_9	
<ul style="list-style-type: none"> <Screen, Error> <Language, English> <Text, 'Payment failed'> 	<ul style="list-style-type: none"> <Text, true> <Symbol, true> <Language, English> <Source, Industriehof> <Destination, Münzenberger straße> <Payment Mode, Cash, Euro> 	<ul style="list-style-type: none"> <Location, Train Station> <Temp, (-20, 50), °C> <Screen Height, 80, cm> 	<ul style="list-style-type: none"> <S_9, (payment_failed), S_1>

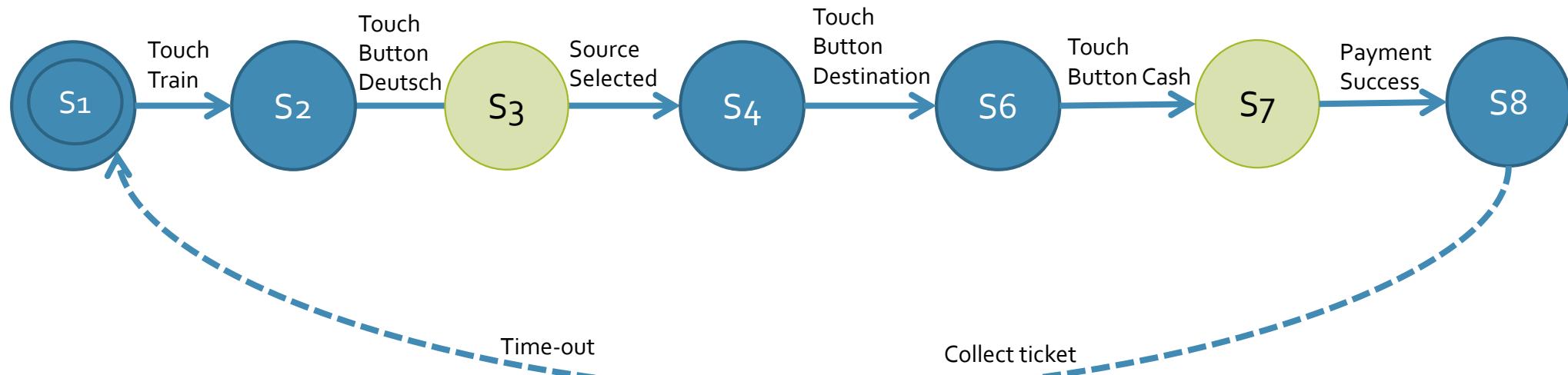
Execution Graph

User: Destination Name Unknown



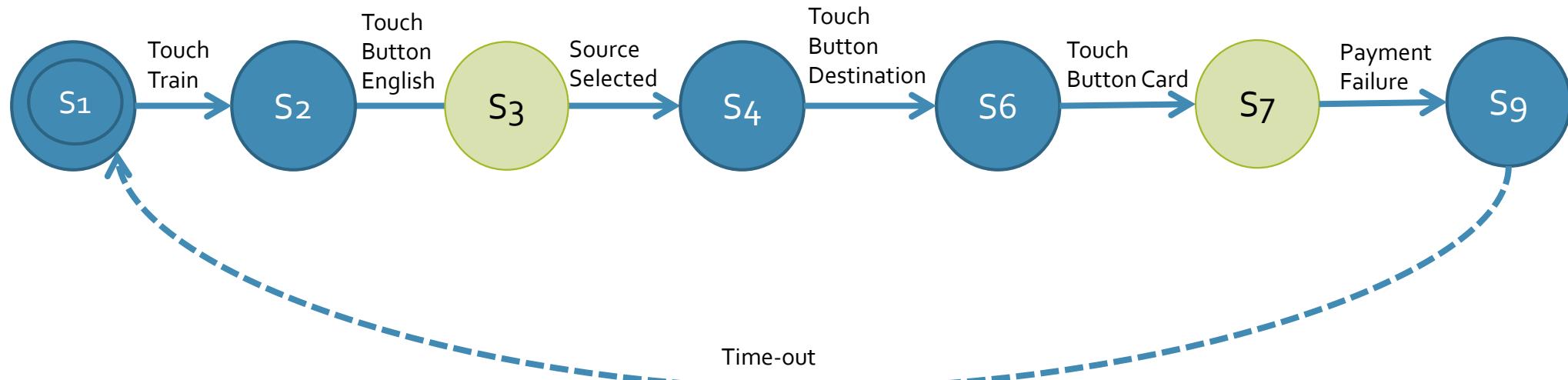
Execution Graph

User: Destination Name known



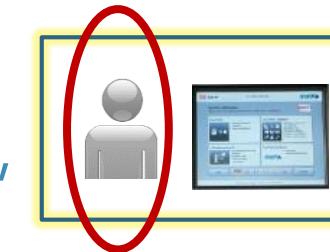
Execution Graph

User: Destination Name known & Invalid Payment Card



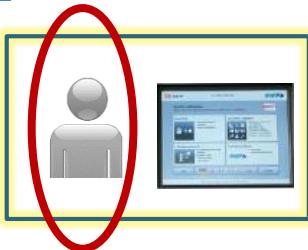
User Model

- Formal State Graphs & User Story
⇒ each state in the User story comprises of 3 sub-states,
i.e. $S_i = \{\text{ENV}, \text{SI}, \text{U}\}$.
- we shall now look closely into the User represented by 'U' in the State Graph.
- User Models (~UMod) [6] –
 - Cognitive
 - Rule-Based [eg: GOMS Model]
- possible User Properties relevant to our example:



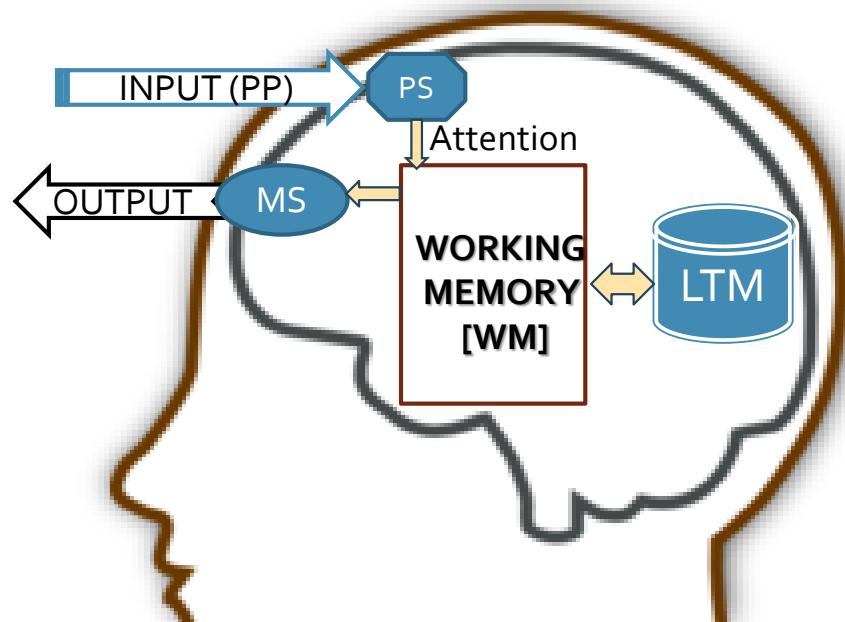
U: User				
<Text, true>	<Symbol, true>	<Language, English>	<Destination, LIDL near FRA-UAS>	<Payment Mode, Cash, Euro>
<Text, false>	<Symbol, false>	<Language, Deutsch>	<Destination, Münzenberger straße>	<Payment Mode, Card>
		<Language, other>		

[6] By Gerd Doeßen-Henisch, Zeynep Tuncer & Louwrence Erasmus. As on June 24, 2016. Human Machine Interaction within Systems Engineering A Primer [Draft Version 0.06]



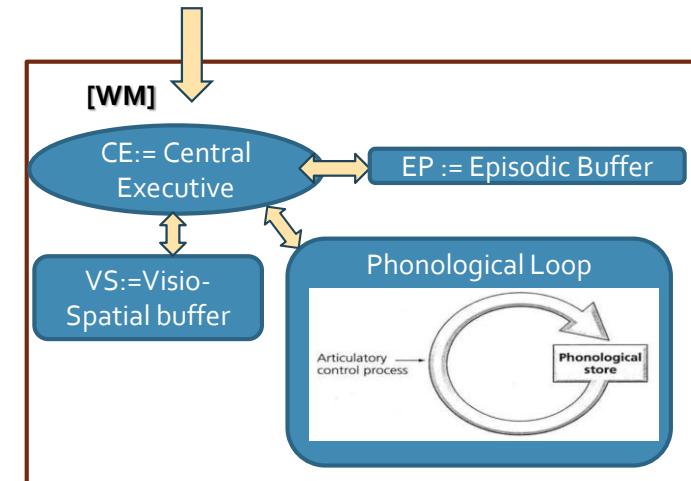
UMod: Cognitive

- The Human Model Processor [4]: an information-processing system.



- Internal States of the Human Processor – [4] [8] [9]

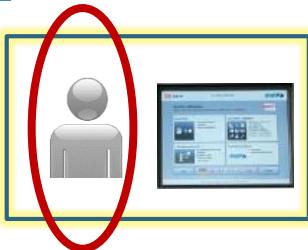
- PS: Perceptual System (Sensory Memory)
 - Visual, Audio, Touch, Smell, Taste..
- CS: Cognitive System
- STM: Short Term Memory
- WM: Working Memory
 - Responsible for cognitive tasks
 - **WM Model x= <CE; VS; EP; PHL>**
- LTM: Long Term Memory
- MS: Motor System



[9] Repovš, Grega, and Alan Baddeley. "The multi-component model of working memory: explorations in experimental cognitive psychology." *Neuroscience* 139.1 (2006): 5-21.

[8] Saul McLeod. Published 2008, updated 2012. URL: <http://www.simplypsychology.org/working%20memory.html>

[4] Stuart K. Card, Allen Newell and Thomas P. Moran. "The Psychology of Human-Computer Interaction". L. Erlbaum Associates Inc. Hillsdale, NJ, USA ©1983. ISBN:0898592437



UMod: Rule-Based

GOMS [5]

- ❑ The GOMS Model is the simplest model that helps in understanding the Human Processor.
- ❑ From the name implies –
 - **G : Goal** >> single High-level Goal G
 >> several Low-level sub-goals Gi
 - [~ G:purchasing a train ticket]
[~ G1:start machine; G2:select language; ...]
 - **O : Operator** >> the set of actions that can be performed by user/human within a certain duration.
 >> can be **perceptual** like 'reading input buffer', **cognitive**(thinking/STM↔LTM) or **motor** like 'touching button'/'collecting ticket'/etc.
 - **M : Method** >> the collective sequence of sub-goals and operators performed towards achieving 1 high-level Goal.
 - **S : Selection Rule** >> the set of rules that direct the user to choose an operation, when there are multiple choices, originating from multiple methods leading to one same goal.
- ❑ In our example of TVM – the user goal **G** is “to purchase a train ticket”
- ❑ To achieve this main goal, there are 6 sub-goals {G₁, G₂, G₄, G₅, G₆, G₈} explained in the following slides.

[5] By Lorin Hochstein. GOMS. October 2002. URL: <http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/printer/goms.html>

UMod: Rule-Based

G1: Start Machine
 Operator: readInputBuffer, icon.train, icon.hand, text

Selection Rule:
 IF (text.known=true || hand.known=true)
 THEN (out=icon.train; s2)
 ELSE (out=readInputBuffer; s1)

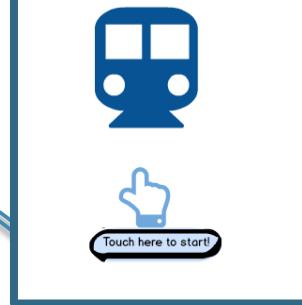
User – English speaker

G2: Select language
 Operator: readInputBuffer, button.English, button.Deutsch, icon.Home

Selection Rule:
 IF (will.abort || will.restart)
 THEN (out=icon.Home; s1)
 ELSE IF (text.eng.known=true)
 THEN (out=button.English; lang=Eng; s3; s4)
 ELSE IF (text.deu.IS=true)
 THEN (out=button.Deutsch; lang=Deu; s3; s4)
 ELSE (out= readInputBuffer; s2)

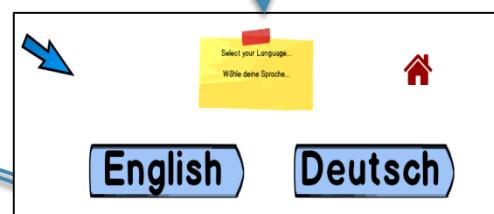
Visual: icon, text

Touch: icon.train



Visual: Text, icon, buttons

Touch: button.English



s1: Home Screen

s2: Language Options

s3: Select Source Station*

UMod: Rule-Based

G4: Select destination
 Operator: readInputBuffer ,
 button.map, text.search,
 button.<station>, icon.Home,
 button.back, button.next

Selection Rule:

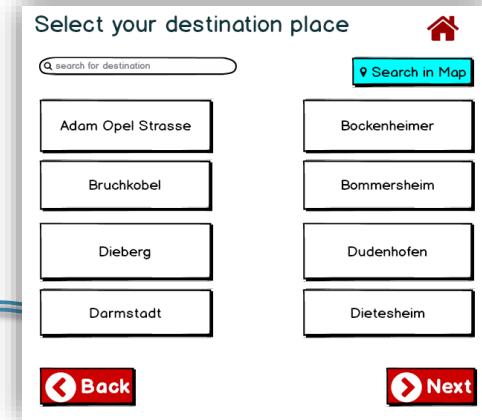
```
destination=null; map=False
WHILE(destination=null && map=False)
{ IF (will.abort || will.restart)
    THEN (out=icon.Home; s1)
ELSE IF (<destination>.unknown)
    THEN (out=button.map; map=True; s5)
ELSE IF (<name>.known && <name>.visible)
    THEN (out=button.<station>;
destination=<station>; s6)
ELSE IF (<name>.known && <station>.find)
    THEN(out=text.search)
    ELSE IF (will.back)
        THEN (out=button.back; s2)
ELSE (out= readInputBuffer; s4) }
```

User – English speaker



Visual: text, icon,
 buttons, search field

Touch: button.<station>



s4: Select
 Destination
 Station

s6:
 Payment
 Screen

UMod: Rule-Based

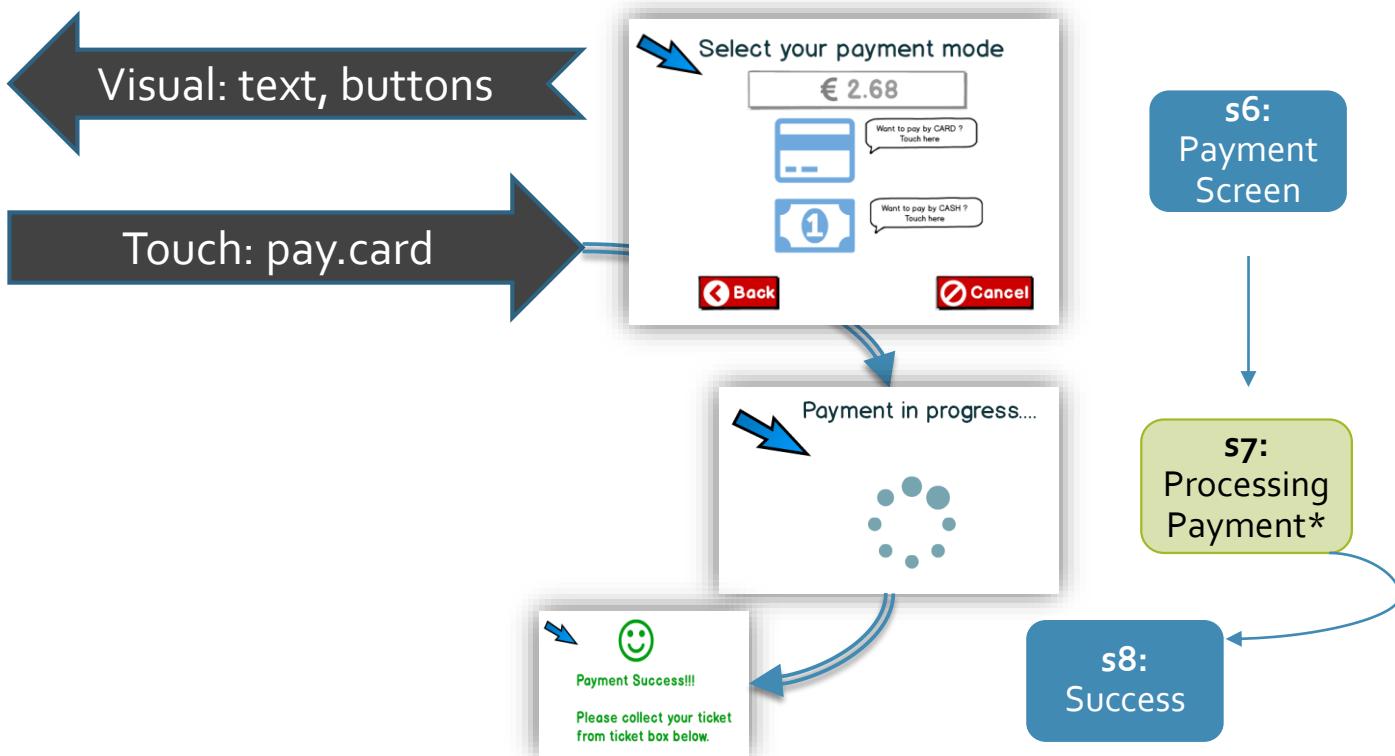
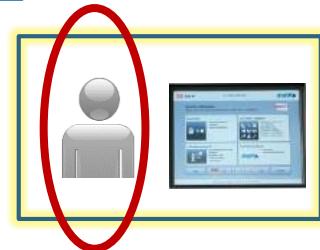
G6: make payment
 Operator: readInputBuffer, pay.card, pay.cash, button.cancel, button.back

Selection Rule:
 IF (will.makePayment) THEN
 (Switch (userChoice) {
 case 1: out=pay.card; break;
 cas2 2: out=pay.cash; break; } s7)
 ELSE IF (will.back)
 THEN (out=button.back; { IF (map) THEN (s5)
 ELSE (s4)}
 IF (will.abort || will.restart)
 THEN (out=button.cancel; s1)
 ELSE (out= readInputBuffer; s6)

G8: complete process & collect ticket
 Operator: none

Selection Rule:
 IF (s8:success) THEN (**user.collectTicket; s1**)
 ELSE IF (s9:failure) THEN(user.restart; s1)
 ELSE (s1 on time-out)

User – English speaker



UMod: Rule-Based

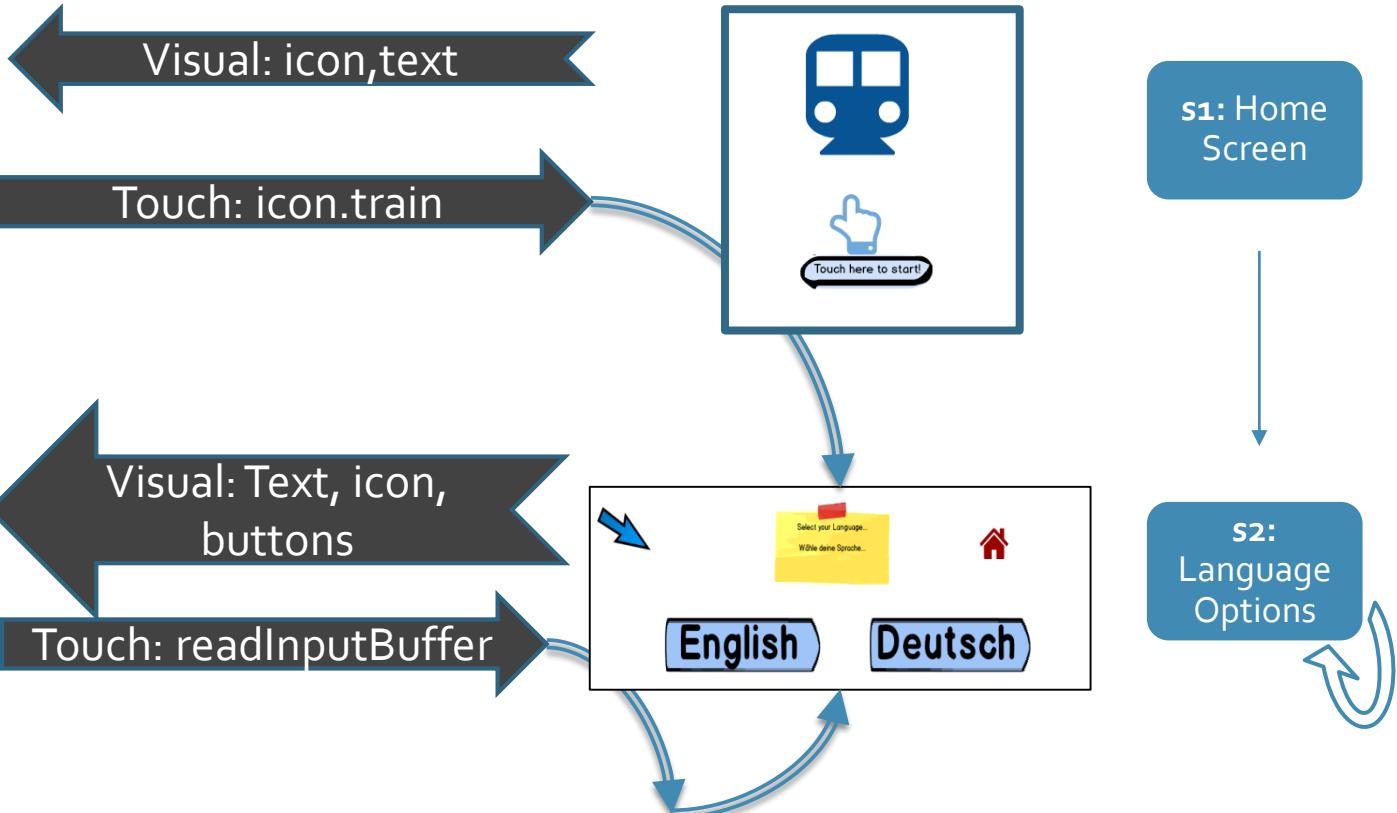
G1: Start Machine
 Operator: readInputBuffer,
 icon.train, icon.hand, text

Selection Rule:
 IF (text.known=true || hand.known=true)
 THEN (out=icon.train; s2)
 ELSE (out=readInputBuffer; s1)

User – no English or Deutsch knowledge

G2: Select language
 Operator: readInputBuffer, button.English,
 button.Deutsch, icon.Home

Selection Rule:
 IF (will.abort || will.restart)
 THEN (out=icon.Home; s1)
 ELSE IF (text.eng.known=true)
 THEN (out=button.English; lang=Eng; s3; s4)
 ELSE IF (text.deu.IS=true)
 THEN (out=button.Deutsch; lang=Deu; s3; s4)
 ELSE (out= readInputBuffer; s2)



UMod: Rule-Based

G1: Start Machine
 Operator: readInputBuffer, icon.train, icon.hand, text

Selection Rule:
 IF (text.known=true || hand.known=true)
 THEN (out=icon.train; s2)
 ELSE (out=readInputBuffer; s1)

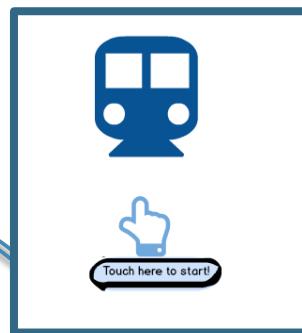
User – Eng, Destination unknown & invalid card

G2: Select language
 Operator: readInputBuffer, button.English, button.Deutsch, icon.Home

Selection Rule:
 IF (will.abort || will.restart)
 THEN (out=icon.Home; s1)
 ELSE IF (text.eng.known=true)
 THEN (out=button.English; lang=Eng; s3; s4)
 ELSE IF (text.deu.IS=true)
 THEN (out=button.Deutsch; lang=Deu; s3; s4)
 ELSE (out= readInputBuffer; s2)

Visual: icon, text

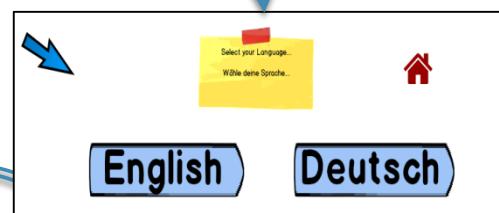
Touch: icon.train



s1: Home Screen

Visual: Text, icon, buttons

Touch: button.English



s2: Language Options

s3: Select Source Station*

UMod: Rule-Based

G4: Select destination
 Operator: readInputBuffer ,
 button.map, text.search,
 button.<station>, icon.Home,
 button.back, button.next

Selection Rule:

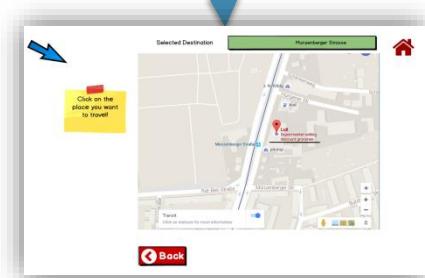
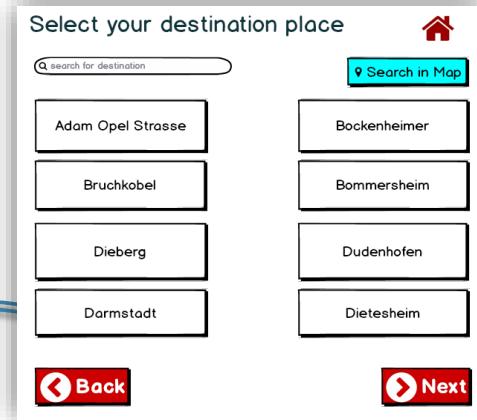
```
destination=null; map=False
WHILE(destination=null && map=False)
{ IF (will.abort || will.restart)
    THEN (out=icon.Home; s1)
ELSE IF (<destination>.unknown)
    THEN (out=button.map; map=True; s5)
ELSE IF (<name>.known && <name>.visible)
    THEN (out=button.<station>;
destination=<station>; s6)
ELSE IF (<name>.known && <station>.find)
    THEN(out=text.search)
    ELSE IF (will.back)
        THEN (out=button.back; s2)
    ELSE (out= readInputBuffer; s4) }
```

User – Eng, Destination unknown & invalid card



Visual: text, icon,
 buttons, search field

Touch: button.map



s4: Select
 Destination
 Station

s5: Map
 View

UMod: Rule-Based

G5: Choose destination

Operator: map.zoom+/-, icon.Home,
map.select, button.<station>,
button.back, readInputBuffer

Selection Rule:

```
destination=null
WHILE(destination=null) {
  IF (will.abort || will.restart)
    THEN (out=icon.Home; s1)
  ELSE IF (<location>.visible)
    THEN (out=map.select;
          out=button.<station>; destination=<station>; s6)
  ELSE IF (<location>.invisible)
    THEN (map.zoom+/-)
  ELSE IF (will.back)
    THEN (out=button.back; s4)
  ELSE (out= readInputBuffer; s5) }
```

Visual: text, button, map, icon

Touch: Map.zoom+/-,
map.select,
button.<station>

User – Eng, Destination unknown & invalid card



s5: Map View



s6: Payment Screen

UMod: Rule-Based

G6: make payment
 Operator: readInputBuffer, pay.card, pay.cash, button.cancel, button.back

Selection Rule:

```

IF (will.makePayment) THEN
  ( Switch (userChoice) {
    case 1: out=pay.card; break;
    cas2 2: out=pay.cash; break; } s7 )
ELSE IF (will.back)
  THEN (out=button.back; { IF (map) THEN (s5)
ELSE (s4)
  IF (will.abort || will.restart)
    THEN (out=button.cancel; s1)
  ELSE (out= readInputBuffer; s6)
  
```

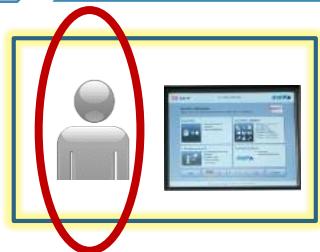
G8: complete process & collect ticket
 Operator: none

Selection Rule:

```

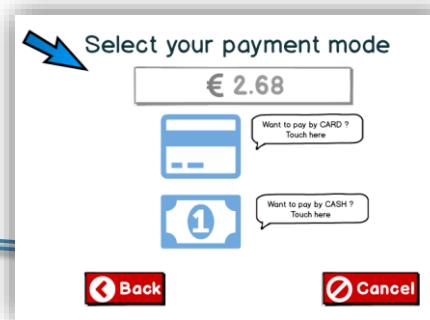
IF (s8:success) THEN (user.collectTicket; s1)
ELSE IF (s9:failure) THEN(user.restart; s1)
ELSE (s1 on time-out)
  
```

User – Eng, Destination unknown & invalid card

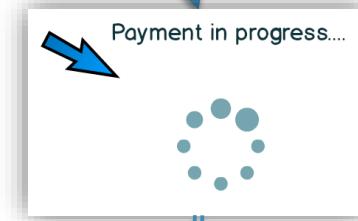


Visual: text, buttons

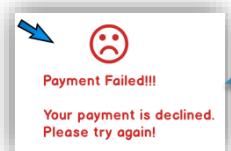
Touch: pay.card



s6:
Payment Screen



s7:
Processing Payment*



s9: Error Screen

Skilled User

$U = \{<\text{Text}, \text{True}>, <\text{Symbol}, \text{True}>, <\text{Language}, \text{English}>, <\text{DestinationPlace}, \text{Nibelungenplatz}>, <\text{Payment Mode}, \text{cash, Euros}>\}$

[SIMULATION]

GOMS USER MODEL

Skilled User:
Input → "T F Eng TRUE Cash"

Program arguments:

T F Eng TRUE Cash

<terminated> GomsModel [Java Application] C:\Program Files\Java\jdk1.8.0_65\bin\javaw.exe (07-Jun-2016, 12:36:00 am)

User Info:	Text	Symbol	Language	Destination	PaymentMode	HOME	GoBack
	true	false	eng	true	cash	false	false

SI-1:HOME Screen {<Icon.train, true>, <Icon.hand, true>, <Text.'Touch here to start!'>}
Input to user: visual{icons, text}

Output: touch{train}

SI-2:LANG Screen {<Button. English, True>, <Button. Deutsch, True>,
<Icon.Home, True>, <Text.'Select a language of your choice'>}

Input to user: visual{icons, text, buttons}
select to change user knowledge -
1) Home=true 2) Home=false 2

Output: touch{button.English}

SI-3:SOURCE is auto selected to current station

SI-4:DESTINATION Screen {<Icon.Home, True>, <Input.Search, True>, <Text.'Select your destination place'>
<button.back>, <buttons.map>, <button.destination>}

Input to user: visual{text, icons, buttons, search field}
select to change user knowledge -
1) Home=true 2) Back=true 3) Home=back=false 3

Output: touch{button.Nibelungenplatz}

<terminated> GomsModel [Java Application] C:\Program Files\Java\jdk1.8.0_65\bin\javaw.exe (07-Jun-2016, 12:36:00 am)

SI-6: PAYMENT Screen {<Icon.Home, True>, <Button.back>, <button.cancel>,
<Text,'Net Payment Amount is 5.20 Euros'>, <Button.card>, <button.cash> }
Input to user: visual{text, icon, buttons}
select to change user knowledge -
1) Home=true 2) Back=true 3) Cancel=true 4) all false 4
Output: touch{button.cash}

SI-7: PAYMENT Processing ...
Your payment through CASH is being processed!!

SI-8: Success

Have a good day !!)

Unskilled User

$U = \{<\text{Text}, \text{False}>, <\text{Symbol}, \text{False}>, <\text{Language}, \text{Hindi}>, <\text{DestinationPlace}, \emptyset>, <\text{Payment Mode}, \emptyset> \}$

[SIMULATION]

GOMS USER MODEL

Unskilled User:
Input → “fail F Hindi F none”

Program arguments:
fail F Hindi F none

User Info:	Text false	Symbol false	Language hindi	Destination false	PaymentMode invalid	HOME false	GoBack false
------------	---------------	-----------------	-------------------	----------------------	------------------------	---------------	-----------------

SI-1:HOME Screen {<Icon.train, true>, <Icon.hand, true>, <Text.'Touch here to start!'>}
 Input to user: visual{icons, text}

textual or symbolic knowledge is mandate!!
 select to change user knowledge -
 1)text=true 2)symbol=true 3)both true 2

User Info:	Text false	Symbol true	Language hindi	Destination false	PaymentMode invalid	HOME false	GoBack false
------------	---------------	--	-------------------	----------------------	------------------------	---------------	-----------------

Output: touch{train}

SI-2:LANG Screen {<Button. English, True>, <Button. Deutsch, True>,
 <Icon.Home, True>, <Text.'Select a language of your choice'>}
 Input to user: visual{icons, text, buttons}
 select to change user knowledge -
 1)Home=true 2)Home=false 2

operational only in english or deutsch
 select to change user knowledge -
 1)lang=English 2)lang=Deutsch 3)lang=other 1

User Info:	Text false	Symbol true	Language eng	Destination false	PaymentMode invalid	HOME false	GoBack false
------------	---------------	----------------	---	--	------------------------	---------------	-----------------

Output: touch{button.English}

SI-3:SOURCE is auto selected to current station

SI-4:DESTINATION Screen {<Icon.Home, True>, <Input.Search, True>, <Text.'Select your destination place'>,
 <button.back>, <buttons.map>, <button.destination>}
 Input to user: visual{text, icons, buttons, search field}
 select to change user knowledge -
 1)Home=true 2)Back=true 3)Home=back=false 3
Output: touch{button.map} 3

SI-5:MAP View {<Icon.Home, True>, <Button.Back, True>, <Map.googleMap, True>,
<Text.'Click on the destination place', <Button.SelectedDestination, True>}
Input to user: visual{buttons,googleMap}
Since we cannot display zoomable map view, destination is auto set.
Output: touch{button.Nibelungenplatz}

SI-6:PAYMENT Screen {<Icon.Home, True>, <Button.back>, <button.cancel>,
<Text,'Net Payment Amount is 5.20 Euros', <Button.card>, <button.cash> }
Input to user: visual{text, icon, buttons}
select to change user knowledge -
1) Home=true 2) Back=true 3) Cancel=true 4) all false 4

2
select to change user knowledge -
1) PaymentMode=cash 2) PaymentMode=card

User Info: Text Symbol Language Destination
 false true eng false

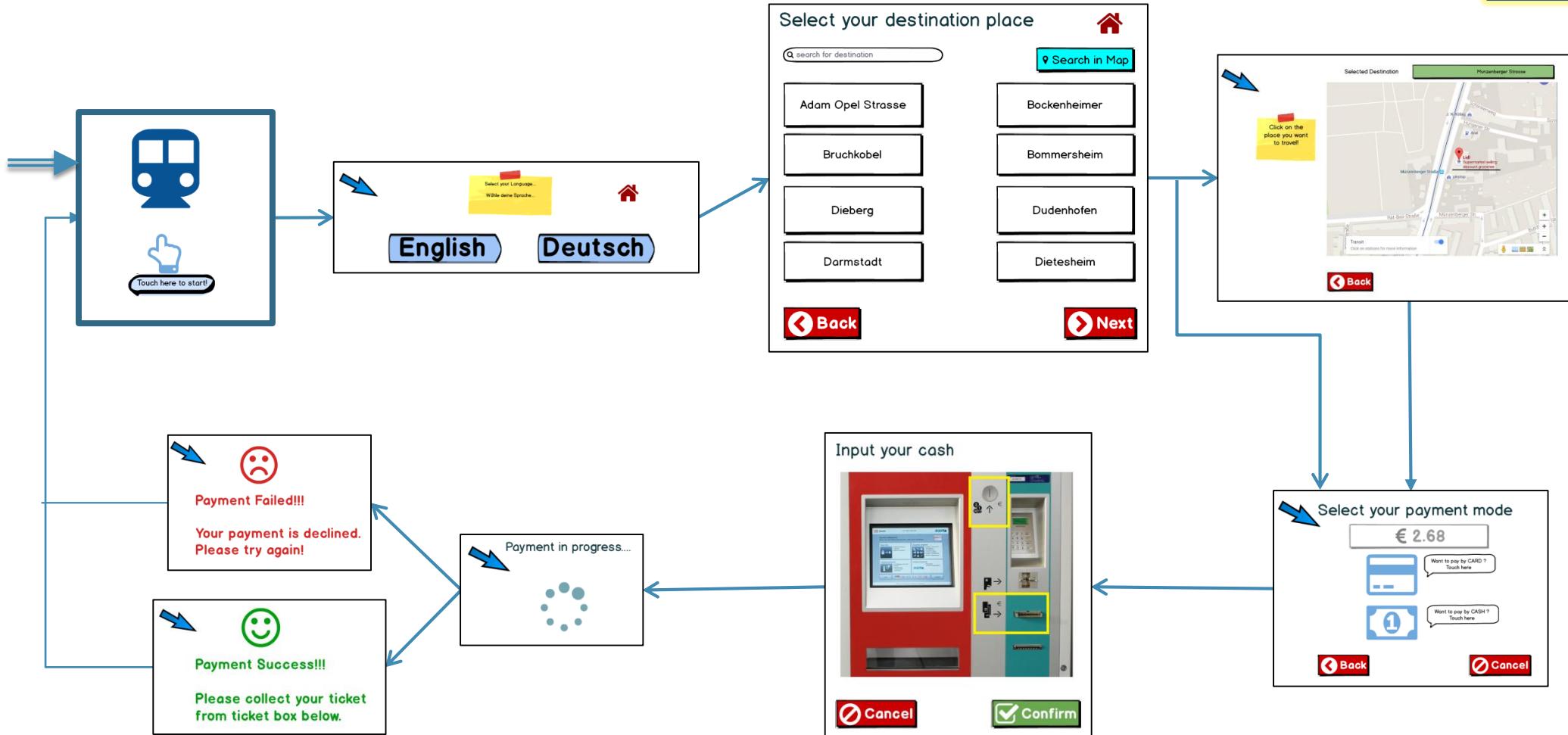
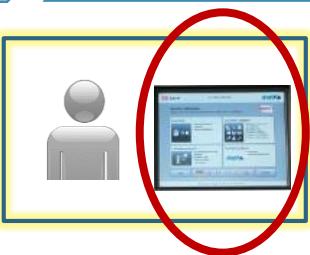
 Output: touch{button.card}

SI-7:PAYMENT Processing ...
Your payment through CARD is being processed!!

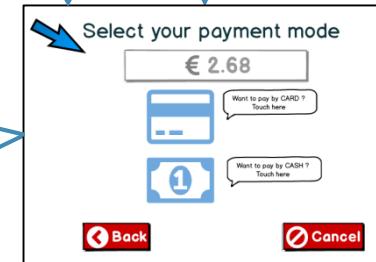
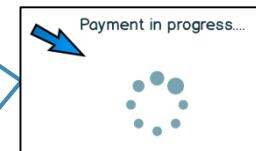
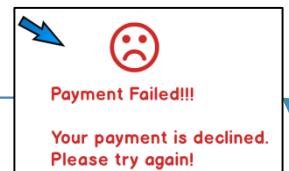
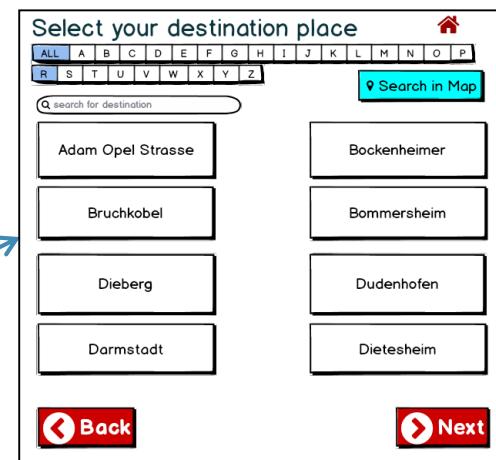
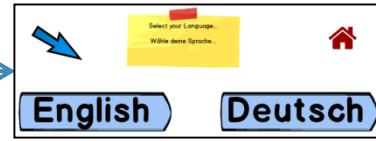
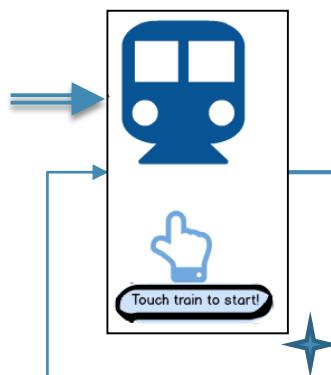
SI-8:Success

Have a good day !!)

1st Mockup



2nd Mockup



modifications

Test Scenario

	ITERATION 1	ITERATION 2
Test Task	purchase a train ticket	
Test Users	Multi-lingual speakers Aware about basics of Automated machines Comfortable using touch screen operations	English/Deutsch speakers
Test Interface	Mouse clickable machine [~ hand touchable screen in real system]	
Test Schedule	To start: 1 min. Task time: 5 min.	
Analyse	Good and bad features about the interface	
Test User count (sample size)	7	3
Evaluators	3	3

Questionnaire

Sl.No.	Usability Components	Questions	# Satisfied Users	
			MockUp 1	MockUp 2
1	<i>Learnability</i>	How easy was it to select destination from map view ?	7	3
2		How easy was the readability in each screen ?	7	3
3		How easy was it to initiate the task ?	5	3
4	<i>Efficiency</i>	How fast were you able to perform the task?	6	3
5		How fast the money transaction process ?	6	2
6		Was all the destination of your choice available ?	2	3
7		Did the system have sufficient language of your choice ?	5	2
8		Is sufficient payment modes of your choice available ?	4	3
9		Was the machine self informative in initiating the transactions ?	7	3
10	<i>Memorability</i>	Were the colours and fonts used in the UI were readable and consistent across all screens ?	4	3
11		Was it easy to navigate between multiple destination screen ?	2	3
13	<i>Minimal Error</i>	All the texts such as messages/tool tip/errors were understandable ?	7	3
14		Were you provided with option to navigate back to right screen in case of wrong action ?	6	3
15	<i>Satisfaction</i>	Rate the overall interaction with the system ?	6	3
16		Rate the map view feature in selecting destination	5	3
			70.54%	89.58%

Heuristic Evaluation [2]

- 1) Visibility of System Status
- 2) Match Between System and the Real World
- 3) User Control and Freedom
- 4) Consistency and Standards
- 5) Error Prevention
- 6) Recognition Rather Than Recall
- 7) Flexibility and Efficiency of Use
- 8) Aesthetic and Minimalist Design
- 9) Help Users Recognize, Diagnose, and Recover from Errors
- 10) Help and Documentation

[2] URL: <http://www.usabilityfirst.com/usability-methods/heuristic-evaluation/>

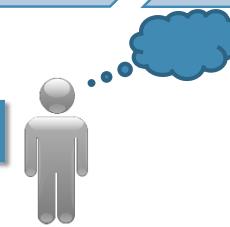
Heuristic Evaluation

Usability Heuristics	Related Question Numbers													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Visibility of System Status	✓		✓	✓				✓		✓		✓		
Match Between System and the Real World				✓		✓	✓	✓			✓	✓		
User Control and Freedom			✓		✓								✓	
Consistency and Standards	✓	✓						✓			✓	✓		✓
Error Prevention													✓	
Recognition Rather Than Recall				✓	✓				✓					
Flexibility and Efficiency of Use			✓	✓										
Aesthetic and Minimalist Design	✓			✓	✓				✓	✓				✓
Help Users Recognize, Diagnose, and Recover from Errors			✓			✓						✓		
Help and Documentation	✓										✓			✓

Heuristic Evaluation

Sl. No.	Questions	MockUp 1	MockUp 2
1	How easy was it to initiate the task ?	71 %	100%
2	Did the system have sufficient language of your choice ?	71 %	97%
3	Was all the destination of your choice available ?	29 %	100%
4	Was the destination easily selectable ?	100 %	100%
5	Was it easy to navigate between multiple destination screen ?	29 %	100%
6	Rate the map view feature in selecting destination	71 %	100%
7	How easy was it to select destination from map view ?	100 %	100%
8	How easy was the readability in each screen ?	100 %	100%
9	Is sufficient payment modes of your choice available ?	57 %	100%
10	Was the machine self informative in initiating the transactions ?	100 %	100%
11	All the texts such as messages/tool tip/errors were understandable ?	100 %	100%
12	Were the colours and fonts used in the UI were readable and consistent across all screens ?	57 %	100%
13	Were you provided with option to navigate back to right screen in case of wrong action ?	86 %	100%
14	Rate the overall interaction with the system ?	86 %	98%
OVERALL %		75.50 %	99.64 %

Thinking Aloud [3]

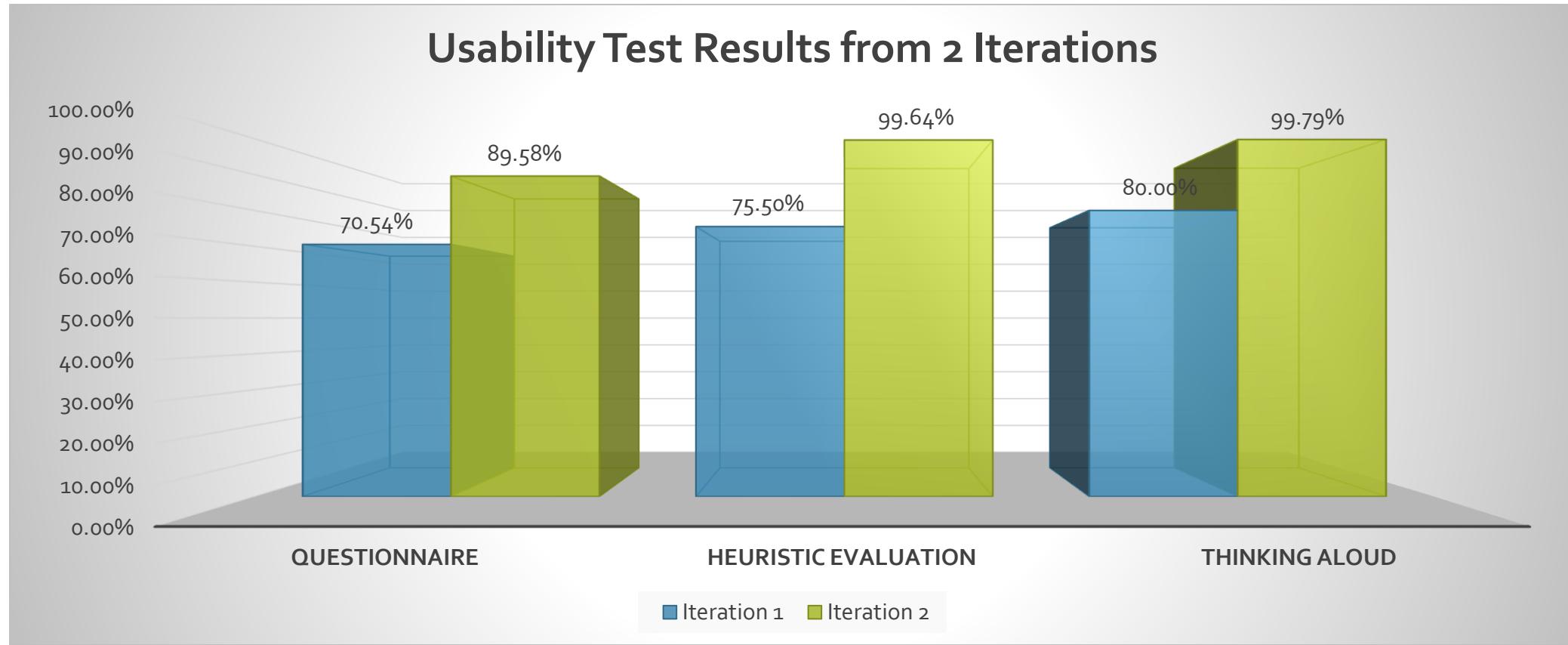


Sl.No.	Questions	User Satisfaction %	
		MockUp 1	MockUp 2
1	How easy was it to initiate the task ?	80%	100%
2	Did the system have sufficient language of your choice ?	90%	98%
3	Was all the destination of your choice available ?	30%	100%
4	Was the destination easily selectable ?	100%	100%
5	Was it easy to navigate between multiple destination screen ?	20%	100%
6	Rate the map view feature in selecting destination	80%	100%
7	How easy was it to select destination from map view ?	100%	100%
8	How easy was the readability in each screen ?	100%	100%
9	Is sufficient payment modes of your choice available ?	70%	100%
10	Was the machine self informative in initiating the transactions ?	100%	100%
11	All the texts such as messages/tooltip/errors were understandable ?	100%	100%
12	Were the colours and fonts used in the UI were readable and consistent across all screens ?	80%	100%
13	Were you provided with option to navigate back to right screen in case of wrong action ?	70%	100%
14	Rate the overall interaction with the system ?	100%	99%
OVERALL %		80.00 %	99.79 %

[3]

By Jakob Nielsen. Thinking Aloud: The #1 Usability Tool. URL: <https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>

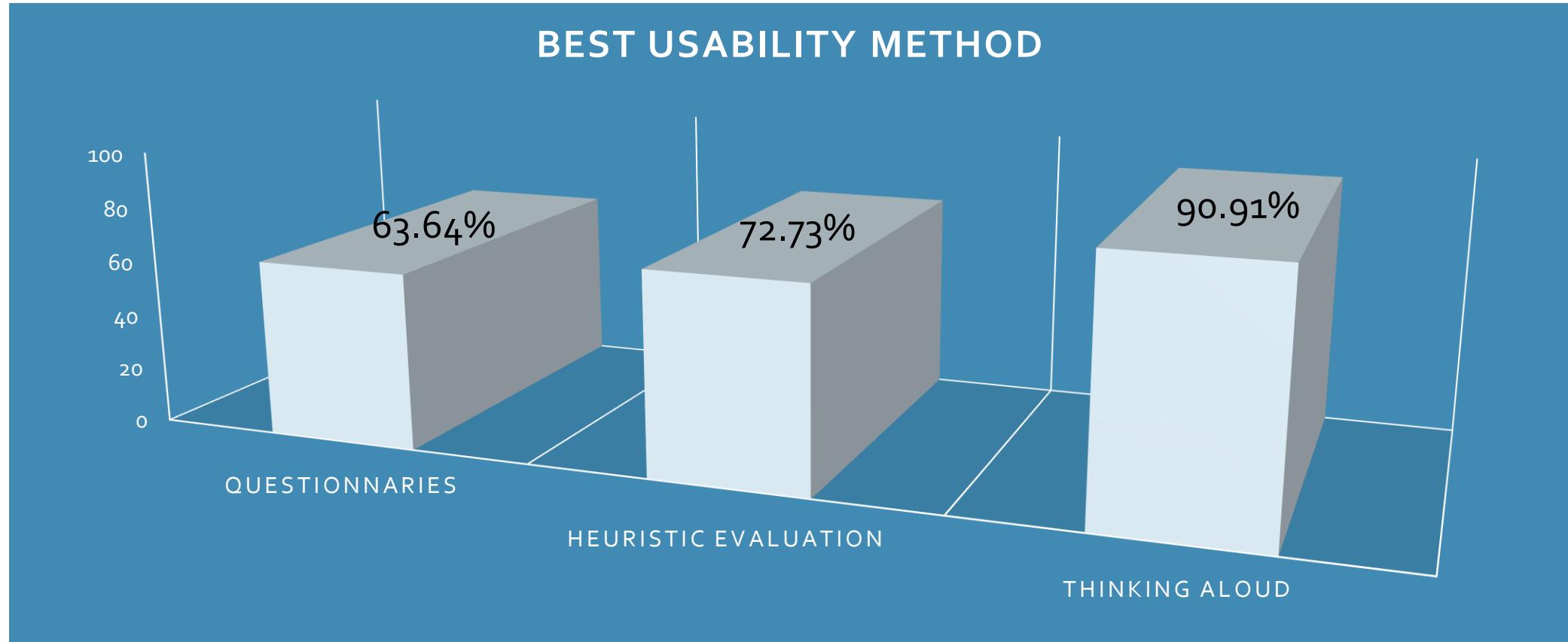
Result Comparision



Method Benefits

	Usability Method [Soft]	Non-Std. Questionnaire	Heuristic Evaluation	Thinking Aloud
Objective measures	Task Time	-	+	+
	Problem Counts	+	+	+
	Keystroke Counts	-	-	+
Subjective measures	Opinion Pole	+	-	+
	Guidelines	+	+	+
	Understanding	+	+	+
Other Properties	Interpretation of result	+	+	+
	Short preparation time	-	-	+
	Quick realizable	+	+	+
	Ref. to weakness	+	+	+
	Low Analysis Time	-	+	-

Method Comparison



Deliverables out of HMI

- Formalized view of the problem task
 - State graphs
 - User Story
 - Automaton; Execution Graph
- Click Dummy, prototype
- Usability Results
- User Model
 - Perceptive & Cognitive user functionalities

Industrial Benefits of HMI

- HMI as a part of Engineering Process helps in achieving
 - Productive Goals
 - Safety Goals
 - Humanization Goals
 - Comprise team and work organization, job satisfaction
 - ergonomic and cognitive compatibility:
 - Environment Goals
 - Refers to the consumption of energy and material resources as well as to impacts on soil, water and air
- Reduces Rework & Improves the quality
- Provides a clear big picture of the problem
- Better End User and Customer Satisfaction

Learnings from the Course

- ❑ Completely different perspective towards approaching a problem
- ❑ How to formalize a problem so that each and every minute properties are taken into consideration
- ❑ How User Model helps
- ❑ Different ways of forming the user stories
- ❑ What are different usability tests and how to evaluate and interpret their results
- ❑ HMI is not all about GUI development
- ❑ How to have a pleasant and healthy discussion while presenting one's thought
- ❑ Best team work and mentors
- ❑ More confidence in facing a real time industrial problem

References

- [1] By Andreas Holzinger. Usability engineering methods for software developers. Communications of the ACM, Vol. 48 No. 1, Pages 71-74. DOI: [10.1145/1039539.1039541](https://doi.org/10.1145/1039539.1039541)
URL: <http://cacm.acm.org/magazines/2005/1/6324-usability-engineering-methods-for-software-developers/fulltext>. Accessed: May 7th 2016
- [2] URL: <http://www.usabilityfirst.com/usability-methods/heuristic-evaluation/>. Accessed on July 9th, 2016.
- [3] By Jakob Nielsen. Thinking Aloud: The #1 Usability Tool.
URL: <https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>. Accessed on July 9th, 2016.
- [4] Stuart K. Card, Allen Newell and Thomas P. Moran. "The Psychology of Human-Computer Interaction". L. Erlbaum Associates Inc. Hillsdale, NJ, USA ©1983. ISBN:0898592437
- [5] By Lorin Hochstein. GOMS. October 2002.
URL: <http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/printer/goms.html>. Accessed on July 9th, 2016.

References

- [6] By Gerd Doeblen-Henisch, Zeynep Tuncer & Louwrence Erasmus. As on June 24, 2016. Human Machine Interaction within Systems Engineering A Primer [Draft Version 0.06]
- [7] By Gunnar Johannsen. CONTROL SYSTEMS, ROBOTICS, AND AUTOMATION – Vol.XXI – Human Machine Interaction.
- [8] Saul McLeod. Published 2008, updated 2012.
URL: <http://www.simplypsychology.org/working%20memory.html>. Accessed on July 9th, 2016.
- [9] Repovš, Grega, and Alan Baddeley. "The multi-component model of working memory: explorations in experimental cognitive psychology." *Neuroscience* 139.1 (2006): 5-21.
- [10] By Avienaaash Shiralige. User Stories: Lack of Big Picture Leads to Blind Man Product.
URL: <http://www.agilebuddha.com/agile/user-stories-lack-of-big-picture-leads-to-blind-man-product/>. Accessed on July 9th, 2016.

Q&A

