



پروژه درس مهندسی نرم افزار

Use Case Description

استاد:

دکتر محمودزاده

اعضای گروه:

صادق اکبری

عرفان رئوفیان

علی غفورزاده

سید مهدی غضنفری

امیرعباس مهدی زاده

Use Case Name: login	ID: 1	Importance Level: High
Primary Actor: User	Use Case Type:	
Stakeholders and Interests: User - Wants to log in.		
Brief Description: This use case describes how the user can log into his account.		
Trigger: User clicks login button		
Type: External		
Relationships: Association: User Include: Extend: Generalization:		
Normal Flow of Events: 1. Enter username 2. Enter password 3. User is logged in		
Sub Flows:		
Alternative/Exceptional Flows: If user enters wrong password: 1. User clicks account recovery button 2. If password incorrect for multiple times: force user to recover the account 3. Else: go to normal flow		

Use Case Name: Sign Up	ID: 2	Importance Level: High
Primary Actor: User	Use Case Type:	
Stakeholders and Interests: User – wants to create an account		
Brief Description: this use case describes how user can create an account		
Trigger: user clicks the sign-up button		
Type: External		
Relationships: Association: User Include: Extend: Generalization:		
Normal Flow of Events: 1. Enters username 2. Enters password 3. Clicks the “done” button 4. Sees successful message		
Sub Flows:		
Alternative/Exceptional Flows: 1. If username not unique: Enters another username again and goes to normal flow of use case 2. If password not strong: Enters another password again and goes to normal flow of use case		

Use Case Name: financial account management	ID: 3	Importance Level: very high
Primary Actor: User	Use Case Type:	
Stakeholders and Interests: User – wants to manage his financial account		
Brief Description: this use case describes how user can manage his financial account		
Trigger: login process triggers this		
Type: Internal		
Relationships: Association: User Include: Add new financial account and category, delete financial account, edit financial account and category. Extend: Generalization:		
Normal Flow of Events: User can either click: <ul style="list-style-type: none">• Add new financial account• Delete financial account• Edit financial account• Add new category• Edit category• Delete category		
Sub Flows: 1. Add new financial account: 1.1. Enters the name of account 1.2. Get account balance 1.3. Clicks on create account 2. Delete financial account: 2.1. Select financial account 2.2. Clicks on delete financial account 2.3. User confirms 3. Edit financial account: 3.1. Select financial account 3.2. Clicks edit account 3.3. Changes old data 3.4. Clicks done 4. Add new category: 4.1. Enters category name		

4.2. Selects category label

5. Edit category

5.1. Selects a category previously created

5.2. Clicks on edit button

5.3. Changes the data

6. Delete category

6.1. Selects a category previously created

6.2. Clicks on delete button

6.3. User confirms

Alternative/Exceptional Flows:

1. Add new financial account:

1.1. User clicks the button "account is a real bank account"

1.2. Enters real bank account number

2. Delete financial account:

2.1. User doesn't confirm

2.2. The account won't be deleted

3. Add new category:

3.1. If the category name user entered already exists system shows error message

4. Delete category

4.1. User doesn't confirm

4.2. The category won't be deleted

Use Case Name: manage financial reports	ID: 4	Importance Level: High
Primary Actor: User	Use Case Type:	
Stakeholders and Interests: User – wants to create report from financial activities		
Brief Description: this use case describes how user can see report of financial activities		
Trigger: user clicks the reports button		
Type: External		
Relationships: Association: User Include: Extend: Generalization:		
Normal Flow of Events: 1. Select one of existing financial accounts 2. Filter categories 3. Select timeline 4. Sees financial report of this account in several statistic forms		
Sub Flows:		
Alternative/Exceptional Flows: 1. If selected account has multiple users: Reports have also information about other users 2. Account has no information considered the applied filters by user: User sees empty information errors		

Use Case Name: transactions management	ID: 5	Importance Level: very high
Primary Actor: User	Use Case Type:	
Stakeholders and Interests: User – wants to manage his transactions		
Brief Description: this use case describes how user can manage his transactions		
Trigger: user clicks “transactions” button		
Type: external		
Relationships: Association: User Include: Add income/outcome/debt/loan, edit income/outcome/debt/loan Extend: Generalization:		
Normal Flow of Events: User can either click: <ul style="list-style-type: none">• Add new income• Edit income• Add new outcome• Edit outcome• Add new debt• Edit debt• add loan• edit loan		
Sub Flows: 7. Add new income: 7.1. Select category 7.2. Enter amount 7.3. Select date and time 7.4. Select related account 7.5. Submit new income 8. Edit income: 8.1. Select one of incomes 8.2. Edit amount 8.3. Edit category 8.4. Edit date and time 8.5. Change related account 8.6. Submit changes 9. Add new outcome: 9.1. Select category 9.2. Enter amount 9.3. Select date and time		

9.4. Select related account

9.5. Submit new outcome

10. Edit outcome:

10.1. Select one of outcomes

10.2. Edit category

10.3. Enter new amount

10.4. Select date and time

10.5. Change related account

10.6. Submit changes

11. Add new debt

11.1. Enter creditor name

11.2. Enter amount

11.3. Enter description

11.4. Select date of settlement

11.5. Submit new debt

12. edit debt

12.1. Select one of debts

12.2. Enter new creditor name

12.3. Enter new amount

12.4. Enter new description

12.5. Change date of settlement

12.6. Submit changes

Alternative/Exceptional Flows: