R E P O R T
Software solution for student housing

ICT & Software Engineering - Semester 1
Class: P-CB-S01
Group: 1
Group members: Stela Trencheva – 4095200
                        Veronika Valeva – 4090349
Date: November/December 2020

Contents:

1. Week one – introduction to the problem
2. Week two – First steps into the solution
3. Week three – Continue the work on our project.
4. Week four – Further expand the application.
5. Week five – Final touch ups

Week One

Work for the week:

On week one we got introduced to the problem and the idea of the project itself. The first thing that we had to do was to create a team that was going to work on the project. We created a group of two and uploaded our group on canvas under number 1. (https://fhict.instructure.com/groups/87650) The next thing in our to do list was to create a GIT repository (https://git.fhict.nl/I451083/p-cb-s01-1-stelatrencheva-veronikavaleva.git) .

After we had done the administrative part our next job was to fully read and understand the problem presented to us. Before giving a start to our project we first had to make a working plan. We made a schedule, distributed the tasks to the group members, and set the deadlines for every deliverable. We made a draft of the project plan and presented it to our teacher. After the consultation, the group members gathered and came up with a better project plan. We decided that one of the group members is going to make the sketches of the pages of the "Student housing BV" application and the other group member is going to make the mock-ups.

By the end of the week (04/12/2020) we have done all the work in the project plan for week one. We are satisfied with the work we have done, and we look forward to the next week.

Week Two

Work for the week:

On week two we began to construct our C# application. The first thing we did was to create a log in page. We decided that the users (the students and the employees) and going to be able to choose their pre-defined profile with their information from a combo box. There are 10 generated students and 5 employees. The students' profiles contain of a student name, age, gender, student number, bank account number and room number. The employees' accounts contain of employee name, employee number, age, gender, e-mails, employee number. All this information is generated from arrays at the beginning of the program. In the log in page, we decided to put two radio buttons for the user to choose whether he/she wants to use a student account or an employee one.

The next thing we had to do was to crate a profile page for the students and the employees. The profile page contains the user's information. On the left side of the application there is a menu with buttons that take you to different user controls. Other than the log in page and the users' accounts, this week we also created the house rules page – for students and employees, the complaints page – for students and employees, the wallet page – for students, the common question page – for students and a pop-up reminder for the undone tasks of the students.

The house rules page for students contains a list box showing all the rules of the building. The house rules page for the employees also has a list box showing the rules. It also has a text box and two buttons. The employee can write a rule in the text box and add it to the list of rules. The employee can also select already existing rule and remove it from the list of rules.

The complaint page for students contains of a tab control with two tabs. The first one is for filling complaints and the second one is for viewing all the complaints. The user can fill out a complaint and add it to the list of complaints. The user can choose to be anonymous with the use of a check box. The complaint page for employees contains a list box showing all the complaints.

The wallet page for students contains the name of the student, his/her bank account number, and the balance in the bank account. The information for the balance in the bank account is also pre-defined in the beginning of the program. There is also a tab control with two tabs. The first one is for viewing the transactions and the overall balance of the certain bank account. The second tab is for deposit or withdraw of money from the student's bank account. In the second tab there are two text boxes – one for inputting the amount of money and the second for describing the reason for the action. The last thing in this page are two buttons – one for withdrawing money from the bank account and the other one for depositing money.

The common questions page contains a combo box with chosen questions – regarding the details of the student's account, a rich text box showing the pre-defined answers for the questions and a button.

We also made a pop-up window to remind students of their undone tasks. At this point we still haven't done the page with the students' tasks and the reminder only shows "You haven't done your chores".

By the end of the week (11/12/2020) we have done all the work in the project plan for week two and we have covered part of the tasks for week three. We are satisfied with the work we have done, and we look forward to the next week.

Week Three

Work for the week:

On week three we have to present a working prototype of our project. We already had a log-in page, account page, a house rules page, a complaint page, a wallet, and a page for common questions.

We created a page where the students can view their tasks. We used tab control to show the undone tasks, the already done tasks and the penalties. There are different types of tasks in "undone tasks" tab. The tasks in this tab are colored in red. There are tasks which the user only selects from the list box and then clicks the button at the button right to do the task and there are tasks which the user first has to select the task and then pay the amount written in the task. A small window pops-up with the amount written on the task, a text box to input your amount of money and a button to finalize your task.

When a task is done it is sent to the other tab where the already done tasks are listed. The tasks in this tab are colored in green. When tasks are done the immediately transfer from the first tab to this second one.

The third tab list all the penalties the student received for not doing his/her chores on time. They are listed in a list box and on the bottom of the form there is a button for paying the penalties. The penalties are colored in red and are listed with numbers. The number of the undone tasks and the penalties are one and the same.

By the end of the week (18/12/2020) we have done all the work in the project plan for week three. We are satisfied with the work we have done, and we look forward to the next week.

Week Four

Work for the week:

The start of the fourth week matched the beginning of the new year – 2021. Throughout this week we finished all the work we had for the project. At the beginning of the week, we discussed how we were going to proceed with the tasks for the week. We decided to try and finish everything by the end of the week as we have even more task for the next week. Throughout those weeks we tried to outjump the timing we had previously set in the project plan. We tried to be a step ahead from the plan and do a little more than needed for each week. Doing so we managed to leave as little work as possible for the last steps of the project period.

The task we have to finish in the fourth week are: Announcement page (for students and employees) – containing three tab controls for different types of announcements (Party, Formal and Covid-19). Common question page for the employee accounts. Manage accounts page for the employees and change room form for the students.

The announcement page for students and employees is the same. It contains tab control – using three tabs for the different types of announcements we have chosen to add. The first and the second tabs are similar in structure. They are about party and formal announcements. They have a list box – showing the already posted announcements, and a group box with controls for creating and adding new announcements. The group box contains radio buttons for selecting the type of the announcement, a text box for selecting the place of the event, a text box for the title of the event, a date-selector, and a button for finalizing the announcement and adding it to the list box of announcements. There are also two other buttons. The first one is called "Restrictions" showing the measures that need to be kept in mind when creating an event. The other button is about deleting an announcement. It works only when an event is selected from the list box. The third tab is about announcements regarding the covid-19. The tab has a list box – showing information about the health of other students in the building. Beneath

the list box there is a button, with which the user can request a test for covid-19. When clicked the user has to select whether his/her test is positive or negative using the radio buttons below the button. After submitting his/her results different actions are undertaken. If the result is positive the user is set under 14 days of quarantine. If the user chooses to make another test and the result is negative the user is automatically removed from the quarantine list.

The common question page for employees has a tab control with two tabs. The first tab is for viewing the questions visible for the students and the second tab is for editing (adding or removing) questions to the list. The "View" tab contains a list box for displaying the questions. The "Edit" tab contains a list box, from with the user can select a question and delete it by pressing the delete button below the list box. This tab also has a text box – for writing new questions and an add button – for submitting and uploading the question to the list box both for the view and the edit tab.

The manage accounts page in the employee accounts is used to view all the students in the application and their information. The page has a list box for showing all the students in the app, from which the user can select a student and view his/her information it the rich text box below.

The change room page for the student accounts is used by student accounts to change their rooms. The page consists of two group boxes. The first one hold the information about the current room of the user and the second group box lets the user choose the information about his/her new room – according to his/her preferences. The user can choose the number of people he/she wants to share the room with and the type of accommodation. Having made his/her choice – the available rooms automatically show in the text box below. The "Book now" button finalizes the process.

At the end of the week, we were left with minor error. We managed to fix all of them in time. By the end of the week (8/01/2021) we were left with minor error. We managed to fix all of them in time. We have done all the work in the project plan for week four. We are satisfied with the work we have done, and we look forward to the next week.

Project development

The first think we had to discuss, before starting with the project, was how we were going to approach the problem. We read the given file with descriptions about the assignment thoroughly and came up with bunch of ideas. We decided that we wanted our application to generate two different kinds of profiles – one for the students – the users of the student housing, and the other kind for employees – the maintainers of the application.

We created 10 different profiles for the students and 5 for the employees. We make collections of gender-neutral names, ages, student numbers, bank account numbers etc. Every time the program is started the profiles are auto generated and the application works with them. We created two classes – Student and Employee. There the student and employee objects are created. The user chooses whether to use the application as a user or as a student.

After logging in the chosen object (selected student/employee) is sent to the "Student/Employee app" form. When the user ends in one of the forms - "Student app" and "Employee app", he/she can see the menu on the right and choose what actions to perform. For this part of the work, we decided to use user controls over different forms for every single window we wanted to implement in the app. We found using user controls to be easier and more user-friendly. We did not want different form popping all the time when changing the different windows of the app. The "Student/Employee app" opens at the "My profile" page. From there the user can choose and go through all the other pages clicking the buttons on the right. When clicked the buttons change colour (all colours, used in the application are color coordinated). On the bottom of the menu the user can find the buttons regarding his/her personal account, details and banking ("My Account", "Change room", "Wallet" (This button is only present for the student accounts), "My tasks", "Common questions", "Manage Accounts" (This button is only present for the employee accounts)). The buttons above regard more common information about the student housing app ("House Rules", "Announcements" "Complaints"). The "Log out" button closes this form and opens the log in form again.

We already described the structures of every single window in the "Student app" or the "Employee app" and the actions it performs in the weekly report at the end of every working week.

The user controls in the "Student app":

- **"House rules"** – Displays all the house rules.
- **"Announcement"** – Allows student user to view/add/remove announcements/events regarding the student housing community. It also has a covid-19 related announcement page. This window was made using tab control to separate the different kinds of announcements.
- **"Complaints"** – Allows student user to view complaints from other student users and add/remove their own complaints. This can also be done anonymously. This was also made using tab control.
- **"My account"** – Shows the main information about the logged student user. There is a button in this form "Change room" with which the user can easily change his/her room.
- **"Wallet"** – Allows the student user to view his/her banking balance and history. The user can also pay his/her penalties/ groceries etc. from here. He/She can also deposit or withdraw money in/from his/her bank account.
- **"My tasks"** – Shows the student user all the task he/she has. They (the tasks) are separated using tab control. ("Done" tasks, "Undone" tasks and "Penalties"). After logging in into a student account a timer starts to tick and after some time a message box pops-up and reminds the user to finish his tasks. If the user decides not to complete his/her tasks they immediately receive penalties, which have to be paid through the "Penalty" window.
- **"Common Questions"** – Show the most common questions regarding the users account and their answers.

The user controls in the "Employee app":

- **"House Rules"** – Allows the employee user to view and edit the house rules.
- **"Announcements" –** This window has the same construction as the "Announcements" window in the student accounts.
- **"Complaints" –** Allows the employee user to view the complaints of the students.
- **"My Account" –** This window has the same construction as the student one, but there is no "Change room" button. There is a button "Download Contract" with which the user can download his/her contract.
- **"Common Questions" –** Allows the employee user the view and edit the common questions for students.
- **"Manage Accounts" –** Allows the employee user to view the list of all students in the application and further see the main info about each one of them.
- **To "Log out" button –** Is the same both for student and employee accounts. When clicked – the student/employee form closes and the "Log in" form opens.

Outcomes

Study outcomes:

Starting with the work on the project we learned a lot of new things from the weekly lessons. All the things we learnt are now part of our application. We came face to face with defining classes and creating objects. We used constructors, methods and properties to create a functional class. We had used methods before but never in the way we did now. We also came across timers, tab controls, random number generator, UML diagrams. We learned how to use multiple forms. We became more familiar with GIT. We still have room for improvements. We spent more time on discussing the way of working and approaching the problem. We created a project plan and it led us throughout those weeks of work. It made things easier for us.

Personal outcomes:

We grew as a team. That is why the personal outcomes regard the team's outcomes. We learnt how to listen to each other and think about each other's ideas. We managed to get all our ideas out in discussion and implement the most successful ones of them. We are satisfied with the outcome and we hope you are too.