# Computer Science Program

3rd Semester 2015

**Semester Project**

**Implementation of Mobile App “Think Tank”**

|  |  |
| --- | --- |
| **The report has been prepared by :** | **Supervisor:** |
| Alexandru Draghi |  |
| Ionut Danci Bumbea | Gianna Bellé |
| Leonard Hirja | Date of submission: |
| Richard Retezi | 14-12-2015 |
|  |  |



# Table of Content

[1. Introduction 5](#_Toc421008958)

[2. Idea Generation 6](#_Toc421008964)

[2.1 POCA Mobile App 7](#_Toc421008965)

[2.2 User Stories 7](#_Toc421008965)

[3 From Idea to Materialization 7](#_Toc421008966)

[3.1 Meeting the Requirements 7](#_Toc421008967)

[3.2 Agile Development and XP 20](#_Toc421008982)

[3.3 Risk Analysis 20](#_Toc421008982)

[3.4 Quality Attributes 20](#_Toc421008982)

[4 Project Implementation 23](#_Toc421008986)

[4.1 Sprint 0 7](#_Toc421008967)

[4.2 Sprint 1 7](#_Toc421008967)

[4.3 Sprint 2 7](#_Toc421008967)

[4.4 Sprint 3 7](#_Toc421008967)

[4.5 Sprint 4 7](#_Toc421008967)

[7 Conclusion 42](#_Toc421009005)

[8. References 43](#_Toc421009006)

[9. Appendices 43](#_Toc421009007)

1. Introduction

Selecting a project that fits the requirements is not hard. There are tons of examples and samples on the internet for many kind of tasks. Selecting a project that is somewhat original, fresh, interesting and challenging is another story. Group 2 decided to go for the latter.

The main requirements for this semester were to code in C#, to have a client/server connection, high complexity, concurrency, and to use some kind of security measures. Regarding to the system development and design, Group 2 had to come up with an idea that can satisfy the above mentioned criteria, create user stories and story cards, design the architecture and interface of the software, plan the whole project, decide on the methodology, the group work and on the schedule.

In the following part the birth of the core idea – inception- is being detailed, as well as how Group 3 managed to develop a project from that single idea.

2. Idea Generation

At the start an idea for the project was needed. The members of Group 2 started to write down ideas on different sheets of papers. After everyone had written down at least 4-5 ideas individually, these ideas were evaluated.

There was one single rule: There are no rules. Group 2 let its imagination fly.

Some ideas were very distant from each other, while others were relatively similar. The ideas were grouped together according to similar thematic and/or platform. After the grouping, each idea was anonymously evaluated. Some of them were discarded instantly, some were kept for future possible consideration or for reserve, while some made the best candidates for the project. A short list was created and in the end, one idea was selected for the project as a common agreement.

In the next section, a short description of the selected idea is presented.

2.1 POCA Mobile Application

The idea was simple: There are lots of young professionals, scattered around the world, who are really interested and/or good in a given area of study and there is a matter they would like to discuss with people with the similar interests. Using modern technology, it would be really nice if they had an app that can find people who share the same passion and provide a field for them to be able to contact and communicate to each other. Using a mobile application they could search through different predefined “passions” and find people of the same interest within their area of living or even anywhere else around the world. When a user finds a person with the same interest he/she can send a request for contact and that person can accept the request. Then, a match is being established and they can start discussing the topics they’d like via chat. In case of multiple people want to discuss a matter, a group chat is possible. This idea can help lots of people who need a platform to share their thoughts and ideas with others who could be interested in the topic.

In the following section an example of a User Stories is shown as example.

2.2 User story

Persona:

Name: Martin Cartridge

Age: 25

Location: Bristol, UK

Occupation: Entrepreneur

Motto: “You cannot know if it’s a failure until you’ve tried”

Story card:

AS A young professional I WANT TO find others with the same area of interests SO I CAN discuss establishing a start-up company with them.

Scenario:

Martin has recently graduated from University and started working in a multinational company. He’s not satisfied with his current position. He’s thinking about launching a start-up company. He has heard about this new app – POCA – from a friend of his. On his way back on the train, he decides to give it a try. He quickly registers by filling some personal information (name, age, location, etc.). He selects 3 of the predefined passions: Entrepreneurship, Economics and IT. He then enters the main page. Martin then searches for people who are also interested in Entrepreneurship. Martin would like to see how many people are interested in this area close to his city, Bristol. He can only find a limited amount of persons – 2 – in his city. Martin then tries to widen the search to see how many people are interested in this Passion in England. He can find significantly more people – 45 --, to connect with. He also tries out a worldwide search to check possible future connections from other countries and he finds – 1700 – users. Martin then chooses 4 persons he could sympathize with and sends a connection request. Martin then waits for his request to be accepted by the other users. After Martin has arrived home, he checks POCA again and sees that 2 users have accepted his request. He writes them a short message, using chat mode. He manages to start a conversation with one of them about how to start a small business.

3. From Idea to Materialization

Vision Statement

FOR people who share the same passion, WHO NEED to connect with like-minded thinkers, POCA IS A mobile app THAT provides the ultimate platform for connecting young professional thinkers. UNLIKE forums, social networks and mainstream communicating platforms, POCA OFFERS an easy and fast way to search and find people within the same area of study and communicate with them.

In this section the process of creating a real-life project from an idea is presented. Now that Group 2 has defined the idea, planning could be started. The first order of business was to check the requirements for the project and align it with the idea.

3.1 Meeting the requirements

The Functional requirements for this project are to use C# programming language, to have a high complexity program, which is able to handle Concurrency, have a database connection and most importantly, to have a client-server connection. The project also has to focus on technology requirements (IT Security measures, protocols, architecture, etc.), and to have a web client that connects to a web server. Most projects are using a website to make the connection, but our idea indicates a mobile application. For the reason that we have to use C#, our platform has to be a Windows Phone that runs on Windows 10 Operating System, which is connected to the WCF (Windows Communications Framework). We need to have a web server connection, therefore we also have to make a sample website for our idea and have a two-sided connection. A Dedicated Client – the mobile app – and the Web Client (sample website). More about the architecture can be read in the Technology Report.

Group 2 also tried to pay attention to Non-Functional Requirements in which the most important features are Usability, Performance, Portability and Security. The characteristic of the project that it is primarily a phone application requires that it is easy for the users to handle, its fast, it can run on multiple phone devices – although because of the requirements (C#), it will run only on windows phone firs – and it has high security to protect the private data of the users and defend against viruses and spam.

3.2 Agile Development and XP

At the start of the project Group 2 had to decide what Development methodology to take: Waterfall or Agile. The project indicates multiple uncertainties, and the group lacks of several key information at the start of the project, therefore a dynamic, flexible approach is needed in order to carefully plan the program. Agile Development is a great way to that, because it lets the members of the group to give a quick, efficient response to unknown occurrences.

Group 2 will use XP (eXtreme Programming), by implementing the incremental delivery approach. This helps the team two break down the projects into several parts, priorities, and ensure that the top priority functions of the program will be shipped and provides possibility for a dynamic response to changes. Some of the functions of our project are significantly more complex and challenging than other – still very important, but less harder to program – functionalities, therefore dividing the priorities into separate iterations/increments proved to be an important task. The most complex function was sure to be coding the search algorithm, there for Group 2 decided to take it over two increments with less complex functionalities (registration, log in).

XP practices were implemented in the project:

* Planning game: User stories and story points were created and evaluated accordingly.
* Small releases: The first release will contain the most important core functions and latter releases will add additional new functions.
* Metaphor: Group 2 decided upon a common framework to work within.
* Simple design: The design for the software will be minimalistic and concentrates on functionality.
* Define Test First: Group 2 designed the architecture and tested the service connection before any coding.
* Refactoring: After the first release and a well-functioned program, design will be revised and improved.
* Pair Programming: PP was introduced which enhanced efficiency.
* Collective Ownership: All the team members could change the software, but only after common agreement.
* Continuous Integration: Group 2 built the program step by step, day by day, exponentially expanding the software.
* 37 hours per week: Group 2 has worked together 3-4 hours and discussed reached milestones, errors and following steps. The group members then worked separately on their own tasks in class or from home another 3-4 hours.
* On-site customer: The product owner is a member of the group, differed in each sprint.
* Coding standards: Naturally, the group implemented many principles and practices to follow and stick with.

Combining XP with Scrum led to a powerful source for planning the project, dividing the iterations into sprints and tasks. The project will consist of 4 sprints (excluding sprint 0) with the implementation of the top 5 priority functions. The different sprints are designed in a way that they equally share a certain amount of tasks which are possible to be completed within the given time period (one sprint should take 1.5 weeks).

3.3 Risk Analysis

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Probability | Effects | Mitigation |
| Prolonged illness (missing out meetings) | Moderate | Serious | Work from Home |
| Technical issues (old laptops with possibility of breaking down) | High | Serious | Make schedule for repairing and upgrading laptop |
| Requirement specifications change original optimal platform selection | Moderate | Serious | Switching to optimal platform |
| Search algorithm proves to more complex than thought before | Moderate | Catastrophic | Change schedule to add more time code Search Algorithm |
| Predefined connection architecture won't work in reality | Low | Catastrophic | Restructure architecture and fix connection errors |
| Potential users are not satisfied with the shipped first release | Low | Catastrophic | Make survey to ask for user reviews to optimize UI and functionality |

In the project Group 2 had to take risks into consideration, therefore it is possible to measure and handle possible faults beforehand and give them a quick and valid response.

3.4 Quality Attributes

Selecting and focusing on some of the many quality attributes are essential to reach a high quality user experience for the program. The main quality attributes the product focuses on are:

* Conceptual Integrity: The software is coherent, consistent in design.
* Maintainability: The program is designed in a way that it is easy to maintain, upgrade, improve or change if necessary.
* Availability: The software is ready to be used by the customers with its core features functioning.
* Interoperability: It is delayed to future implementation, primarily it runs on Windows Phone because of C#. It is planned to be able to run on Android and iOS phones.
* Manageability: It is easy for the administrators to manage the software and intervene in case of errors, viruses, spam attacks, harmful users, etc.
* Security: As probably the most important feature, security protects the users’ personal data and exchange of information.
* Scalability: As the project includes the possibility that many users can register within a short amount of time, the system has to be able to handle a sudden increase of data flow (load) without setback in performance.
* Usability: It is extremely important for Group 2 that the project meets the requirements, implements the core functionalities in its first release and it is easy to use for the users which will result in positive user feedbacks.

Naturally, all the other quality attributes were taken into consideration as well, such as Reusability, Performance, Reliability, Supportability and Testability.

4. Project Implementation

Group 2 drew mock-ups for the project idea, thought about the architecture and collected user stories in order to have a deeper understanding of the project. After the members collected all the possible user stories they could think of, story points were added to them, so that prioritization will be possible. A constant discussion within the group – when deciding on priorities – is that whether the most complex user story the top priority or the one which is simpler, but vital to the project. When deciding on story points, the most complex user stories received the highest score, but that doesn’t necessarily mean that those became the top most priorities, but the ones that are vital.

Within this framework, the group had selected the top 5 priorities user stories and broke it down into tasks. The top 5 priorities were the following: Register and Login (1st), Search (2nd), Request and Match (3rd), Receive and Send message (4th) and Book an Advisor (5th). Even though Search received the highest score because of its complexity (19 pts in total – with subtasks), Register and Login became the top most priority (with 13 story points), because without it the application cannot exist and it is essential for registering and protecting user information. Search received the highest score because the estimated time for writing the search algorithm considered to be the most complex and challenging among rest of the user stories and it is one of the core functionalities of the app. Match and Messaging (or Chat) are also core functionalities and Book an Advisor was important to include because that provides the Concurrency (in order to align with the requirements). Originally another user story was set as 5th priority: Group Chat which allows the users to communicate with multiple other users and share their ideas. Regarding to the fact that the project had to align with the concurrency requirement, Group Chat was moved from the First Release and Book an Advisor took its place.

4.1 Sprint 0

Sprint 0 is the initialization of the project, where all the planning, architecture related decisions and design related solutions are explored. Group 2 had drawn a mock-up for the project as well as a basic architecture in order to have a better understanding of the planned software. A decision was made on which diagrams are needed to help the team in the progress.

The selected diagrams were the following: Use Case Diagram, Domain Model, Candidate Class Diagram, System Sequence Diagrams and Design Class Diagram. These diagrams are important in order to visualize connections and relations in the program.

Group meetings and stand up meetings were also declared during Sprint 0. The team decided to meet at UCN’s Sofiendalsvej campus at 8.30 when there’s class in the schedule, otherwise to meet at Leonard’s and Alex’s dormitory where there is a study room perfectly fitting for the group’s needs and it is the closest place to work on the project for the most of us. The meeting time at the dormitory was set to 11.00. The group had divided the upcoming tasks, decided on the Scrum Master and on the Project Owner for each sprint, and then started to prepare for Sprint 1.

As a kickstart to the project, Group 2 created the Product Backlog which contains all the user stories, even the ones that are not in the top 5 priority. Then the team created the Sprint Task Table and prepared putting the user stories and their related tasks for Sprint 1. A scheme for Sprint Backlog was also created, so the members can the amount of remaining workhours for each task in each user story per each sprint. The values were calculated in a way, that the group decided on 8 hours / day of work in which 4 hours are for coding only, whilst the other 4 hours are for writing the report, discussing problems and also populate and modify are charts and tables (such as the sprint backlog itself). These charts were created in digital format, as it is easier for the members to input, change data and share files through GitHub independently from an exact physical location.

The group also created some physical charts for representation. Sprint Task Board, Release Burn Down Chart and Sprint Burn Down chart were created for that purpose. Sprint Task Board consists of the following columns: User Stories, Tasks, To Do, In Progress, Test (test phase) and Done columns. The tasks (which are on sticky notes) are moved between the columns of the board accordingly. The Sprint Burn Down Chart represents the correlation between the amount of story points and the time (number of days) spent on working per sprint. Release Burn Down chart is aggregating all of the Sprint and it shows the total sequence in one single chart.

After all the preparation, the team was ready for Sprint 1.

4.2 Sprint 1

Sprint 1 consisted of two main areas: Setting up the database and making the connections with the software via the two primary user stories: Register and Login. Register and Login were selected as the first two user stories for sprint 1 because even though they are pretty simple, they are indispensable and the program cannot start/function without them. Creating the database, setting up the empty tables and queries were relatively simple as the group also had experience with these tasks. Creating the interface for register and login was also simple and was done within one day.

As mentioned before, the target platform was Windows Phone 8.1. As the team had experience on developing small applications (games) on windows phone before and also had experience working with databases, the members thought that after a short period of research on architectures (connections between databases and windows phone), it will be possible to connect the database with our chosen platform. Unfortunately the group had faced some serious obstacles and challenges.

The team decided to make a test first in order to test the connection. The result of the test was negative, meaning that even though the separate elements of the program (the classes, the model layer, the database, the interface, etc.) were programmed well the connection itself failed. Group 3 was stuck on this problem for two and a half days. The connection was tested multiple times with slight changes and failed all over again. The members decided to step back, get a fresh air and think on the problem. The team revised the classes, the database set ups and the architecture. The team revised how connections and the settings of the connection needed to be done between the databases, the software and the target platform. The program was tested on the phone and also on an Emulator (Visual Studio’s windows phone 8.1 emulator). Because the project requirements include a segment which is to create a web application (connect to a web server and to a website), the members decided to test the project if it is working on a web app.