5 Project Information

5.1 Key Stakeholders

The Key Stakeholder for the project as per Appendix B [12] – The case study is Alena. She is the owner of the health care center based in Geelong. Alena is a chiropractor by profession and wants to expand the property to accommodate a podiatrist and a naturopath to the healthcare center. Alena is also the main investor in the business and will fund this business, as stated in the case study [12] where she has the capital on renovating and expanding the healthcare center.

The other important stakeholder identified for the project is Gregor. Gregor is Alena’s brother and a Student in business and IT at The University of Melbourne. He is well versed with the software system and will be an important link between the development team and Alena.

5.2 Scope

5.2.1 What is in-scope?

According to the Case Study, The SWEN90016 [12] has to develop a web based appointment system for Alena’s healthcare center. From the Key requirement given, the team has identified that there will be 9 webpages for the project.

1. **Login Page** – Used for logging into the system. There are two users login to the system are

a) The Admin (based on key requirement 1)

b) Customers (based on key requirement 4)

The login would require an email id and a password, which would be predefined for admins but needs to register for customers. The login page would also have register link where new customers would register.

1. **Admin Page** – when an Admin logs in , we navigate to the admin page , here there are two tabs-
2. Add\_healthcare\_profession
3. View\_appointment\_request\_admin
4. **Add\_healthcare\_profession page**- Admin navigate to this page to enter the detail of healthcare professional value like Type, Name, and Email address, consultation charges are captured and saved.
5. **View\_appointment\_ request\_admin page** – Admin navigates to this page would view the aggregated information of all the booking history captured from appointment booking page and information sent to the health care profession via email. This information would be a tabular format where the admin can view the data.
6. **Customer Registration page** (personal\_information) - This page pops up If the customer clicks on the new user link below the login page. Information such as Name, Home address, contact phone number, Email ID and password is captured in this page. Once you click on save you are redirected to the login page.
7. **Customer page** – The customer page has three tabs where they can navigate to.
8. Update personal\_information page
9. Appointment\_booking page
10. View Appointment\_booking page
11. **Update personal\_information page** - customer navigate to this page to edit the information they enter in the registration page, once the customer make changes and they are redirected to the login page.
12. **Appointment\_booking page –** In the key requirements, the customers are allowed to select the health professional from the list of available options or select the health professional by name for appointment booking. Once this is selected the health care professional’s available time, charges are populated. The customer has to choose the appropriate available time between the available hours. There is also an option to send a message to the health care professional. After you click submit, an appointment is booked.
13. **View\_Appointment\_booking page** – When the customer navigates to this page they can view or delete the booking information. All the information related to the appointment booking page will be stored in this page in tabular format. {This will not be developed as we are given concession for this point}.

Also, we will be using a database to store customer, healthcare professional and appointment booking information for future use in the project.

Finally, after submitting or deleting an appointment booking, an on-click trigger will mail the health professional with Name, phone number, and email address of customer along with date and time of the booking and message which is optional.

5.2.2 What is out-of-scope?

There are few things which are out of scope in the requirement phase, which could be added in future enhancements

1) As mentioned in Appendix B, Healthcare professionals don’t have the option to log in to the system.

2) In the personal information page, there is no requirement for password recovery. Generally when you register we confirm the password twice and there is always password a recovery question like what is your mother’s name. Which is missing in the requirement. So if the customer forgets the password there is no way to recover that.

3) The case study[12] states that there can be only three medical professionals at a time, but as per the requirement we want to capture the historical data of all healthcare professionals working there. So there is no way currently in the system to detect which are active medical professional available.

4) In the view\_appointment booking page, the customer is able to view or delete the booking. There is no option to edit or modify the booking. In the current situation, the customer will have to delete and make a new booking but it would be easier to add the modify functionality.

5.3 Delivery approach

SDLC - Formal or Agile ☐ Formal ☐ Agile ☐ Hybrid

In a software project selecting the appropriate software development lifecycle is the most important step as it greatly increases the overall project success. Selecting the right model can decrease development cost/time, reduce overheads, risk exposure and most importantly help better project tracking and control. From the case study [12] the team feel agile scrum methodology would work best for the project. There are few reasons why agile would work the best here [1**4**][13]:

1. The project is web based and the project size is relatively small, which works perfectly in the scrum software development model.
2. The scrum team consists of people with cross-functionality experience which greatly overhead costs given that the budget is less.
3. This model gives Alena better opportunity to pivot the requirements as the requirements are not well defined and would need changes.
4. The scrum model has ability to integrate and scale better than the traditional model. As the business grows after it becomes successful, it’s easier to add new features or functionality if we are using agile model as they can be added on the backlog and developed later on.

5.4 Business Value (Financial & Non-Financial Benefits)

The biggest benefit for the main stakeholder Alena is convenience and time. In the current model, a lot of time and effort is wasted in extracting the booking information and Alena is always busy gathering that either in her lunch break or end of the day. Also, a huge effort is wasted when a new customer calls and Alena has to follow up on the phone. So the system gives a nice structure to day to day working where the existing customer can log in and book their appointment and a new customer would register and then log in and book.

The second business value the software system provides is it creates a backbone for a potential platform where multiple locations and multiple health care profession can be added. This scaling is possible as the business grows and becomes successful

The business value to Gregor is this software system gives him real-world experience in software project management. As Gregor is business and IT student this experience is valuable and would help him in his career.

5.5 Constraints

The constraints identified are budget and I.T experience of the stakeholders. It’s well documented in the case study [12] that the team will be working on a shoestring budget as most of the money was used in renovating and expanding the center. Also, the stakeholder Alena and Gregor have very little I.T and project management experience between them. Alena is chiropractor by profession and does not have any software development experience

Gregor is a student, he may know the I.T concept and might speak in technical terms but its unknown what his experience is. So it will be a little difficult for the scrum master and development team to understand the requirement and later translate that to appointment management web-based system.

6.5 Project Planning

For this case study [12], the team will be using the agile scrum methodology. So to execute this case study we will be using various scrum tools to get out of every cycle. The Scum tools used are The Scrum product backlog, the release burndown chart, the sprint backlog, and the scrum task board.

**The scrum product backlog:**

This is a prioritized features list, containing short descriptions of all functionality desired in the appointment management system [17]. The product backlog is based on the key requirement given by stakeholders.

In the Scrum product backlog, we have identified item types like user story features and technical requirement. The user story is identified from key requirements and technical requirement [17]. Bugs and knowledge acquisition have not been identified as of now as the project is in the primary stage and they will be added later subsequent versions.

|  |  |  |
| --- | --- | --- |
| **Product backlog List** | **Description** | **Type** |
| Database connection and query | Initial database connection | Technical requirement |
| Login page | This is the most important page as both the admin and customers login through this page | User story |
| Personal\_information page | New customer can register and later login. | User story |
| Add Health care profession page | Admin can add doctors and doctor information in this page | User story |
| Appointment\_booking page | Customers can book appointment through this page | User story |
| Email trigger | Email trigger to the healthcare professional once the customer book an appointment | User story |
| View\_Appointment\_booking page | It contains the list of appointment for customer can view or delete appointment | User story |
| Email trigger | Email trigger to the healthcare professional once the customer cancel an appointment | User story |
| View\_appointment\_request\_admin page | admin can view the list of all the appointment by customer | User story |
| Update personal\_information page | Customer can update personal information in this page | User story |
| Customer page | Customer page has tabs for updating personal information, appointment booking page and view appointment booking page | User story |
| Admin Page | Admin page has tabs for Add healthcare profession page and view appointment request page | User story |
| Database connection and query | stores customer, healthcare professional and appointment\_booking information | Technical requirement |

Table 6 : scrum product backlog [17]

**The sprint backlog**

In the sprint backlog [15], the scrum master identify the list of tasks which would be completed during a scrum sprint. The team looked at the product backlog items and identified task for each item and tasks which should be needed to complete each at sprint story. The team is using user story points as a unit of time for this case study [12].

Since the project size is small, each sprint cycle duration would be one week. Team has identified that the team can complete 25 user points per week as a rough estimate with roughly 10% positive or negative impact on this estimation.

Below table is the sprint backlog table, in this table we have identified user story based on product backlog table and subdivided into smaller tasks which need to be completed in the product backlog table. User points are added for each task and the total points for each user story.

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Tasks | Relative user points | Total user point per task |
| Database connection | Check database connection | 1 | 2 |
| Test Connection | 1 |
| Login page | Design the webpage | 1 | 6 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| on click code for admin login | 0.5 |
| on click code for dummy customer login | 0.5 |
| on click code for new personal\_information page | 0.5 |
| Automate test for the on click code | 1.5 |
| Personal\_information page | Design the webpage | 1 | 5 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| on click code for save | 0.5 |
| Automate test dummy customers | 1.5 |
| Add Healthcare\_profession page | Design the webpage | 1 | 5 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| on click code for save | 0.5 |
| Automate test dummy health professional | 1.5 |
| Appointment\_booking page | Design the webpage | 1 | 8 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| code to search healthcare professional by type | 1 |
| code to search healthcare professional by name | 1 |
| appointment booking time and date | 0.5 |
| Automate test to search healthcare professional by type | 1 |
| Automate test to search healthcare professional by Name | 1 |
| Automate test the booking time and date | 0.5 |
| Email trigger | Code the email trigger to send appropriate information about the customer to healthcare professional when appointment is booked | 1 | 1.5 |
| Automate test | 0.5 |
| View\_Appointment\_booking page | Design the webpage | 1 | 4.5 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| On click code to delete | 0.5 |
| Automate test to delete appointment | 0.5 |
| Email trigger | Code the email trigger to send appropriate information about the customer to healthcare professional when appointment is deleted | 1 | 1.5 |
| Automate test | 0.5 |
| View\_appointment\_request\_admin page | Design the webpage | 1 | 4 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| populate all the appropriate data | 0.5 |
| Automate test that the data is populate properly | 0.5 |
| Update personal\_information page | Populate the saved data from appointment\_booking page | 0.5 | 1 |
| Automate testing once the data is saved | 0.5 |
| Customer page | Design the webpage | 1 | 5.5 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| on click code for update personal\_infomation page | 0.5 |
| on click code for View\_appointment request page | 0.5 |
| on click code for View\_appointment request page | 0.5 |
| on click code for Appointment\_booking page | 0.5 |
| Automate testing for all the on click functionality | 0.5 |
| Admin Page | Design the webpage | 1 | 4.5 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropriate changes | 0.5 |
| on click code for Add Healthcare\_profession page | 0.5 |
| on click code for View\_appointment\_ request\_admin | 0.5 |
| Automate testing for all the on click functionality | 0.5 |
| Database connection and query | Check database connection | 0.5 | 1 |
| run query for customer, healthcare professional and appointment booking data | 0.25 |
| Test the output with manually | 0.25 |

Table 7 : scrum product backlog [15]

During the scrum sprint, we add new information to the sprint backlog. Based on that the scrum master plots a sprint backlog chart. This chart keeps on updating throughout the project. We have plotted the sprint backlog chart based on our initial requirement.

Based on the above table we have 49 user story in total and would break them down into two sprint cycle of one week. In the first sprint cycle, we would complete 26 sprint point and 23 in the second sprint point.

The items in the sprint cycle are based on priority, weight and functional flow based on dependency. The main aim is to make a minimum viable product which will allow the stakeholder to start her business. We have identified database connection, Login page, Personal\_information page, Add Healthcare\_profession page and Appointment\_booking page in the first sprint cycle.

Below is the table which denote the first sprint cycle item. It includes the user story, the tasks and the user points for that task.

|  |  |  |
| --- | --- | --- |
| User story | Tasks | Relative user points |
| Database connection | Check database connection | 1 |
| Test Connection | 1 |
| Login page | Design the webpage | 1 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropiate changes | 0.5 |
| onclick code for admin login | 0.5 |
| onclick code for dummy customer login | 0.5 |
| onclick code for new personal\_information page | 0.5 |
| Automate test for the onlick code | 1.5 |
| Personal\_information page | Design the webpage | 1 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropiate changes | 0.5 |
| onclick code for save | 0.5 |
| Automate test dummy customers | 1.5 |
| Add Healthcare\_profession page | Design the webpage | 1 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropiate changes | 0.5 |
| onclick code for save | 0.5 |
| Automate test dummy health professional | 1.5 |
| Appointment\_booking page | Design the webpage | 1 |
| Design the UI | 1 |
| Meet Alena and get approval for UI-UX design | 0.5 |
| make appropiate changes | 0.5 |
| code to search healthcare professional by type | 1 |
| code to search healthcare professional by name | 1 |
| appointent booking time and date | 0.5 |
| Automate test to search healthcare professional by type | 1 |
| Automate test to search healthcare professional by Name | 1 |
| Automate test thebooking time and date | 0.5 |

Table 8 : first sprint cycle [17]

**Burndown chart**

The burndown chart [18] is an essential part of any agile project and is a way for the team to clearly see what is happening and how progress is being made during each sprint. In our Sprint cycle, the horizontal axis of the sprint burndown chart shows the sprints; the vertical axis shows the amount of work remaining at the start of each sprint. Work remaining can be shown in whatever unit the team prefers—story points.

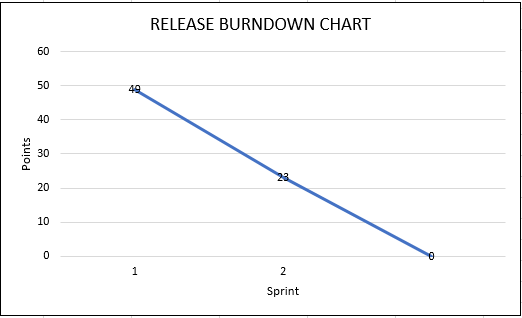


Figure 1: Release burndown chart [18]

Based on the above document for the first sprint cycle, we aim to complete 26 sprint point and in the subsequent future sprint, we aim to complete 23 sprint point.

**The scrum task board**

During the sprint cycle, the sprint backlog [16] is made visible by putting it on a Scrum task board. Team members update the task board continuously throughout the sprint for e.g. if someone thinks of a new task, they writes a new card and puts it on the wall. During or before the daily scrum, estimates are changed (up or down), and cards are moved around the board.

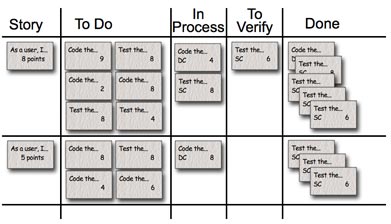
[](https://www.mountaingoatsoftware.com/uploads/articles/MockedTaskBoard.jpg)

Figure 2: Scrum taskboard [16]

Each row on the Scrum board is a user story which is based on the requirements in our case study [12] from product backlog.

During the sprint planning meeting, each product backlog item is turned into multiple sprint backlog items which is represented by one task card that is placed on the Scrumboard. Each task card starts on the Scrum taskboard in the “To Do” column [16].

The columns we would use on our task board are:

* Story: The story description shown on that row.
* To do: Place for all cards that are not in the “Done” or “In Process” columns for the current sprint.
* Work in process: Any card being worked on goes here. The programmer who chooses to work on it moves it over when she's ready to start the task.
* To verify: A lot of tasks have corresponding test task cards.
* Done: Cards pile up over here when they're done. They're removed at the end of the sprint. Sometimes we remove some or all during a sprint if there are a lot of cards.

Once the task is complete we built the minimum viable product, which is something which is the minimum development we make so that the running of the website can begin. Also after every sprint cycle, we show our progress to the stakeholder and have an internal meeting where we can discuss what went right and what went wrong and how will we improve to make the subsequent sprints better.

References

[13]"A comparison of scrum and Kanban for identifying their selection factors", *Ieeexplore-ieee-org.ezp.lib.unimelb.edu.au*, 2017. [Online]. Available: https://ieeexplore-ieee-org.ezp.lib.unimelb.edu.au/document/8312434?arnumber=8312434&SID=EBSCO:edseee. [Accessed: 05- Apr- 2019].

# [14]” SCRUM model for agile methodology, Ieeexplore-ieee-org.ezp.lib.unimelb.edu.au, 2017. [Online]. Available: https://ieeexplore-ieeeorg.ezp.lib.unimelb.edu.au/document/8229928?arnumber=8229928&SID=EBSCO:edseee. [Accessed: 01- Apr- 2019].

[15]M. Cohn, "Sprint Backlog and the Scrum Sprint", *Mountain Goat Software*, 2019. [Online]. Available: https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/sprint-backlog. [Accessed: 24- Apr- 2019].

[16]M. Cohn, "Scrum Task Board", *Mountain Goat Software*, 2019. [Online]. Available: https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/task-boards. [Accessed: 24- Apr- 2019].

[17]M. Cohn, "Scrum Product Backlog and Agile Product Backlog Prioritization", *Mountain Goat Software*, 2019. [Online]. Available: https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/product-backlog. [Accessed: 24- Apr- 2019].