**AGILE PROJECT SPRINT 1**

**CLINICAL MANAGEMENT SYSTEM**

**GROUP 3**

**Due: 16 November, 2018**

**Members**

**Aman, Semir - 101142654**

**Chaudhry, Ayusha - 100968449**

**Naguit, Sir Angel - 101152749**

**Victa, Francis Bertholet - 101159185**

**Yang, Maria Lilian - 101151657**

**Table of Contents**

1. **Project Charter3**
2. **The Client System and User Personas** **5**
   1. ABC Clinic System **5**
   2. Personas **6**
3. **User Stories** **10**
4. **Prioritized Features List 12**
   1. Prioritized List **12**
   2. Feature Definitions **13**
5. **Prioritized Product Backlog** **15**
6. **Implementation Plan 26**
7. **Project Charter**

Project Name: Clinical Management System

Vision: Help the clinic to convert its paper-based system into an easy to use and robust software system that stores all information electronically.

Mission: Create an online database which enables the staff to access patient records in real time, schedule appointments, and generate reports.

Success Criteria:

1. Implement database by January 2019

2. Conduct more than 80% or greater of paper-based work via the online system

3. Increase efficiency and accuracy to 100%

4. Finish the project in a 4 month period.

Project Team

|  |  |
| --- | --- |
| **Name** | **Role** |
| Angel Naguit | Developer |
| Maria Lilian Yang | Scrum Master |
| Francis Victa | Product Owner |
| Semir Aman | Developer |
| Ayusha Chaudhry | Developer |

Rules of Behavior:

1. Treat each other with respect and courtesy

2. Open communication including regarding constructive positive and negative feedback

3. All individual and team achievements are valuable

4. Everyone will be held accountable for their actions

5. Work together in a team in a collaborative manner

Communications:

1. Team meetings take place every morning at 9am in the meeting room

2. Update tasks on the backlog daily based on progress during meeting

3. All team members are expected to be on time for all meetings

Specific Roles:

The Scrum Master helps each developer in every sprint before going to the next Sprint.

The Scrum Master devotes 50% of his/ her job to developing and 50% towards his/her role as Scrum Master. If any of his/her tasks overlaps another, he/she prioritizes Scrum Master task before the other.

The Product Owner guides the teams towards achieving the client’s needs of the system. He/she becomes the key to success in every feature test to check whether it is suited to client’s current expectation such as to its design and interface.

1. **Client System and User Personas**
2. **The ABC Clinic System**

ABC is a walk in clinic with ten (10) family doctors, eight (8) nurses and twelve (12) receptionists. Some of the clinic staff work part time.

The clinic is open Monday to Friday from 9-8 pm with day shift running from 9-5 pm and an evening shift 5-8 pm.

The Clinic has 2 kinds of patients, patients with family doctors who can book an appointment with them and walk in patients. Default duration for every patient with the doctor is 15 minutes and 30 minutes for special cases.

The Patient Check-in/out Process:

1. Patient's walk-in and requirements such as health card, patient address, and contact details are checked by secretary where status becomes "checked in".

2. If patient needs urgent care based on the triage system, he/ she will be added on top of the queue line.

3. A nurse takes the patient from the queue to an examination room and takes his temp, height, weight, bp, history of problems/symptoms.

4. Nurse notes it on the patient's file updates patient's file.

5. The nurse hands over this file to the doctor.

6. Doctor examines the patient and updates patient file with symptoms, diagnosis, medications or lab tests prescribed.

7. Sometimes doctor refers the patient to a specialist doctor and enters this information in the file.

8. Patient is charged with sick notes fee when needed and patient file is updated about the fee and payment status

9. At the end of the appointment, the secretary marks a “check out" status. In case of abnormal results patient is called back to the clinic by the doctor.

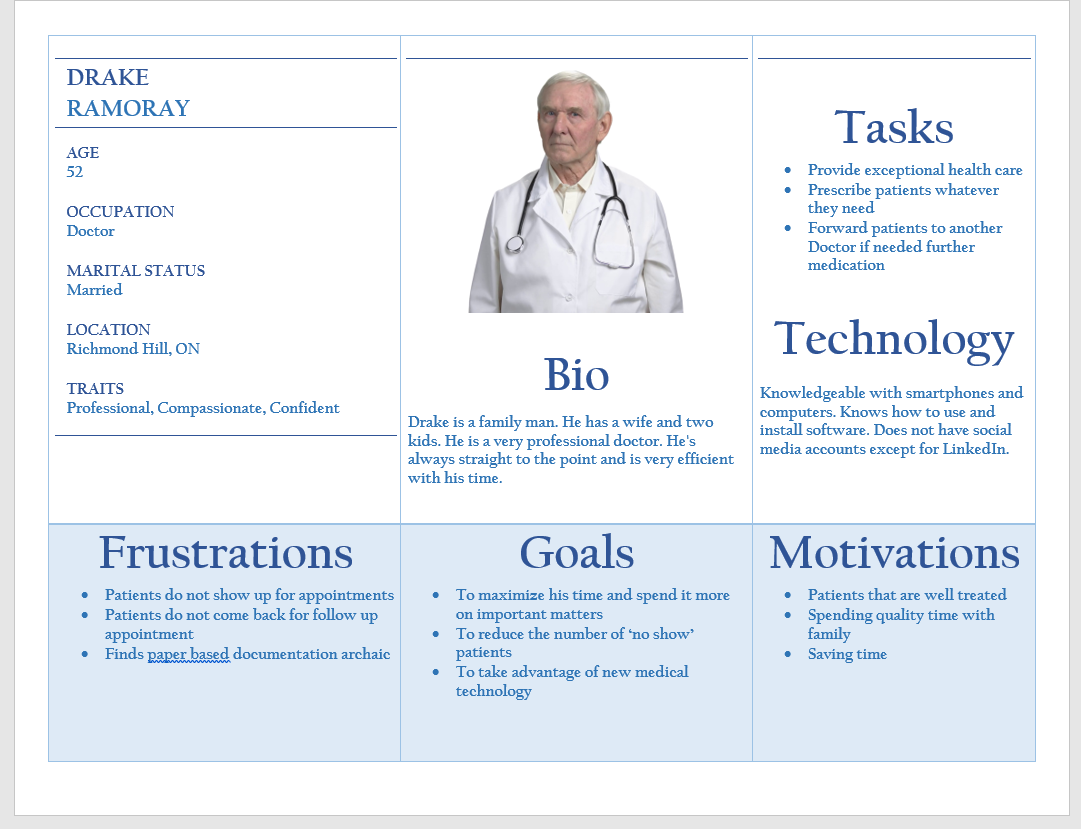
1. **Personas**
   1. **User Persona: Receptionist**

****

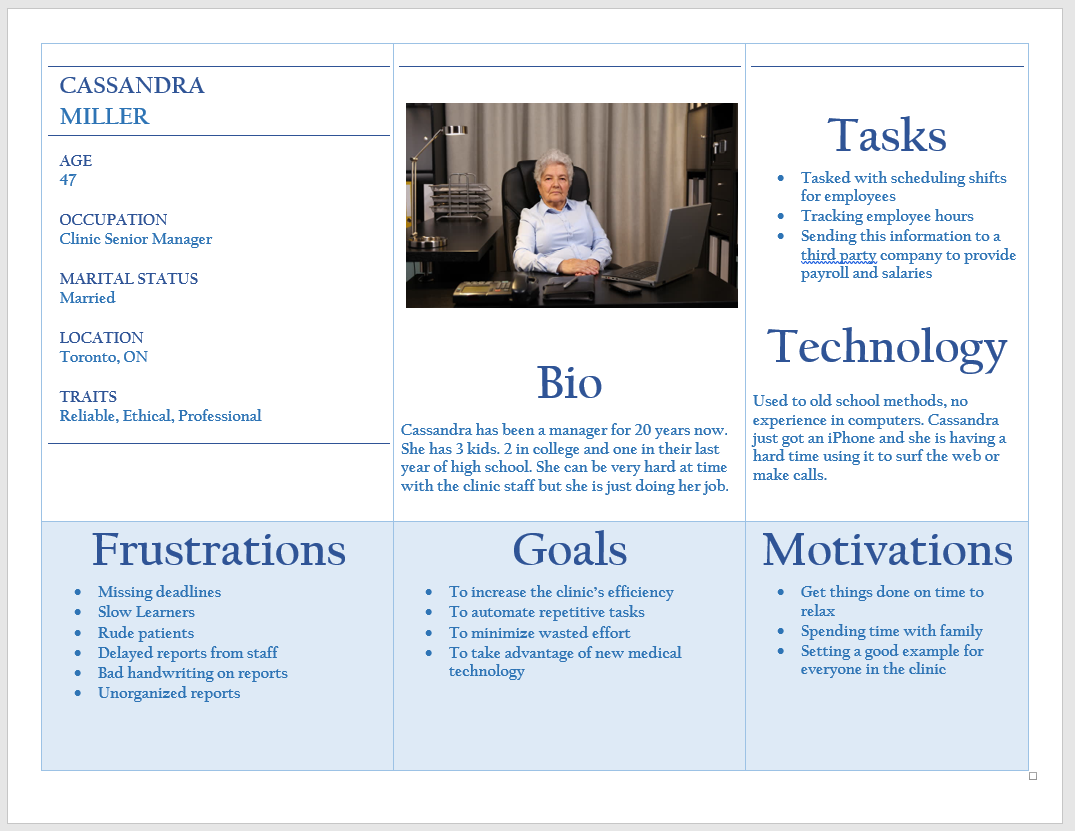
* 1. **User Persona: Nurse**

****

* 1. **User Persona: Doctor**

****

* 1. **User Persona: Manager**

****

1. **User Stories**

|  |  |  |  |
| --- | --- | --- | --- |
| **Persona** | **Priority** | **Size** | **Stories** |
| **Doctor** | must have | small | As a doctor I want to search a patient's file to be able to access it. |
| must have | small | As a doctor I want to write on a patient's file so I can update patient's symptoms, diagnosis or medications that I prescribed |
| could have | small | As a doctor I want to forward a patient's file to a specialist doctor to easily perform a referral |
| **Nurse** | must have | small | As a nurse I want to search a patient's file so that I can access their information |
| must have | small | As a nurse I want to write on patient's file so that I can update their vital signs and biometric records |
| **Receptionist** | must have | small | As a receptionist I want to view a patient file so I can verify their health card, address and contact details |
| should have | small | As a receptionist I want to update a patient file so that their records are up to date |
| must have | small | As a receptionist I want to create a new patient file so that new patients will be stored in the system |
| should have | small | As a receptionist I want to check-in a patient file so they can be added to the queue |
| should have | small | As a receptionist I want to check-out a patient after their appointment so that their file gets updated |
| should have | big | As a receptionist I want to add patients to the queue so that the queue list is organized |
| should have | big | As a receptionist I want to remove patients from the queue so that the queue list is updated |
| should have | big | As a receptionist I want to move a patient anywhere in the queue so that patients who need urgent care are taken care of immediately |
| must have | uncertain | As a receptionist I want to cancel a patient's appointment so that the schedule will free up for the doctor |
| must have | big | As a receptionist I want to book a patient for an appointment for their family doctor so that they can see their doctor |
| should have | big | As a receptionist I want to flag patients on file who do not show up for their appointment so that the system can inform the patient of the $30 fee |
| should have | small | As a receptionist I want to flag patients who request for sick notes so that it will be recorded in their file |
| **Manager** | must have | big | As a manager I want to schedule shifts for my staff so that they would know their schedule |
| should have | big | As a manager I want to schedule reports to automatically send to my email so that I have an update of 'no show' and 'special service' fees |
| must have | small | As a manager I want to receive hours worked report for all staff every two weeks so that I can forward the information to our accountant |
| could have | big | As a manager I want to generate reports on the system so that I can use it to help with my clinical analysis |
| must have | uncertain | As a manager I want to add new staff to the system so that they will be part of the system |

1. **Prioritized Features List**
2. **Prioritized List**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Priority** | **Priority Rank** | **Size Estimation** | **Story Point** | **Feature** |
| must have | **1** | small | 4 | Add a new patient file feature. |
| must have | **2** | big | 6 | Search a patient file |
| must have | **3** | big | 6 | View a patient file |
| must have | **4** | big | 8 | Book an appointment |
| must have | **5** | uncertain | 7 | Cancel an appointment |
| must have | **6** | big | 4 | Schedule Shifts |
| must have | **7** | small | 3 | Update a patient's file |
| must have | **8** | small | 2 | Get the total number of hours in a week |
| must have | **9** | uncertain | 4 | Add a new staff member |
| should have | **10** | big | 8 | Check-in a patient to queue |
| should have | **11** | big | 8 | Check-out a patient in a queue |
| should have | **12** | big | 10 | Add to Queue List for emergency or urgent triage feature |
| should have | **13** | uncertain | 10 | Remove from Queue List for receptionist |
| should have | **14** | big | 6 | Adding a "no show" fee to a patient who missed an appointment |
| should have | **15** | small | 4 | Add a Sick Note fee to a patient file |
| should have | **16** | big | 6 | Generate General Reports for the Manager |
| should have | **17** | uncertain | 8 | Send Email Feature |
| should have | **18** | uncertain | 8 | Open a received email feature |
| should have | **19** | small | 4 | Login Feature for staff members |
| should have | **20** | small | 4 | Logout Feature |
| could have | **21** | big | 8 | Scheduled Reports Feature |
| could have | **22** | big | 8 | Easy Referral feature from patient file to a Specialist doctor |
| could have | **23** | big | 8 | Queue Status |

**B. Feature Definitions**

|  |  |  |
| --- | --- | --- |
| **Priority Rank** | **Feature** | **Feature Definitions** |
| **1** | Add a new patient file feature. | The receptionist should be able to create a new patient file. |
| **2** | Search a patient file | The receptionist, nurse and doctor should be able to search for a patient file |
| **3** | View a patient file | The receptionist, nurse and doctor should be able to search for a patient file |
| **4** | Book an appointment | The receptionist should be able to book an appointment for a patient to their family doctor |
| **5** | Cancel an appointment | The receptionist should be able to cancel an appointment 24 hours after the patient have booked if requested by the patient to be cancelled. |
| **6** | Schedule Shifts | The manager should be able to add staff member shifts schedules |
| **7** | Update a patient's file | The nurse, receptionist, Dr should be able to update file for patient's biometrics and patient history. |
| **8** | Get the total number of hours in a week | The manager should be able to get the total number hours each employee has incurred for weekly reports |
| **9** | Add a new staff member | The manager should be able to add a new staff member. |
| **10** | Check-in a patient to queue | The receptionist should be able to add patient to a queue list. |
| **11** | Check-out a patient in a queue | The receptionist should be able to remove a patient to a queue list. |
| **12** | Add to Queue List for emergency or urgent triage feature | The receptionist should be able to add an urgent patient on top of the list. |
| **13** | Remove from Queue List for receptionist | The receptionist should be able to remove a patient from the queue list. |
| **14** | Adding a "no show" fee to a patient who missed an appointment | The receptionist should be able to add a fee to patient's file for patients who do not show up for their appointment |
| **15** | Add a Sick Note fee to a patient file | The receptionist should be able to add a sick note file and fee to patient's file for patients who requests for one |
| **16** | Generate General Reports for the Manager | The manager should be able to generate reports from the database based on various filters and parameters |
| **17** | Send Email Feature | Every staff member should be able to send private messages to a specific staff member for transfer of patient info |
| **18** | Open a received email feature | Every staff member should have the ability ro receive a private message from any staff member for transfer privacy of patient info |
| **19** | Login Feature for staff members | Every staff member should be able to use the system through the logging in the system using their credentials |
| **20** | Logout Feature | Every staff member should be able to log out after using the system for security purposes |
| **21** | Scheduled Reports Feature | The manager should be able to schedule specific reports such as weekly "no show fees" and hours worked by the staff to easily manage the reports. |
| **22** | Easy Referral feature from patient file to a Specialist doctor | The doctor should be able to send a referral to a specialist through the system |
| **23** | Queue Status | The nurse should be able to mark the queue status of the patient as "with the doctor" to show in the queue line that patient is being seen by the doctor and to inform the other nurse to call next patient in the queue line |

1. **Prioritized Product Backlog**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task number** | **Task Name** | **Duration** | **Predecessors** | **Assigned** |
|  | **Sprint 1** |  |  |  |
| 1 | Sprint planning, design and analysis | 0.5 |  | Programmer A,B,C, ScrumMaster, Product Owner, Client |
|  | **Design and Implementation of the Clinic Database** |  |  |  |
| 2 | Upgrade Visual Studio software on Teams' laptops to add DotNet module for ASP.NET database connectivity and Prepare Visual Studio for C# graphical interface | 1 | 1 |  |
| 3 | Team members take full day course on ASP.NET | 1 | 2 | Programmer A, B, C, ScrumMaster, Product Owner |
| 4 | Design database structure | 2 | 3 | Programmer A, B, C, ScrumMaster, Product Owner |
| 5 | Design and create Patient database | 0.5 | 4 | Programmer A |
| 6 | Design and create Staff database | 0.5 | 4 | Programmer B |
| 7 | Design and add Staff Schedule table to Staff database | 0.5 | 4 | Programmer C, Product Owner |
| 8 | Design and create Queue table to Patient database | 0.5 | 5 | Programmer A |
| 9 | Design and create Appointment table to Patient database | 0.5 | 6 | Programmer B |
| 10 | Design and create Transactions table to Patient database | 0.5 | 7 | Programmer C, Product Owner |
| 11 | Design and create Email database | 0.5 | 9 | Programmer B, ScrumMaster |
| 12 | Design and create a Reports database | 0.5 | 10 | Programmer C, ScrumMaster |
| 13 | Check and review the Database | 1.5 | 8,11,12 | Programmer A,B,C, ScrumMaster, Product Owner |
| I | **Add new patients** |  |  |  |
| 14 | Create 'Add Patient' method which adds new Patient file in the Patient database | 1 | 13 | Programmer A |
| 15 | Create a graphical interface for the receptionist to Add Patient details | 1 | 13 | Programmer A, ScrumMaster |
| 16 | Connect 'add Patient' method to add Patient graphical interface | 0.5 | 14,15 | Programmer A |
| 17 | Test method and fix bugs found | 2 | 16 | Programmer A, Product Owner |
| 18 | Integrate and final test or fix the methods inside the current build of the system | 1 | 17 | Programmer A |
| **II** | **Search for a Patient File** |  |  |  |
| 19 | Create 'Search' method that retrieves data on the list of Patients from the patient database | 1 | 13 | Programmer B |
| 20 | Create a graphical interface for all Staff members to Search Patient details | 1 | 19 | Programmer B, ScrumMaster |
| 21 | Connect 'Search' method to search graphical interface | 0.5 | 19,20 | Programmer B |
| 22 | Test method and fix bugs found | 2 | 21 | Programmer B, Product Owner |
| 23 | Integrate and final test or fix the methods inside the current build of the system | 1 | 22 | Programmer B |
| **III** | **View Patient File** |  |  |  |
| 24 | Create 'View Patient File' method that updates data on the Patient database | 1 | 18 | Programmer C |
| 25 | Create graphical user interface of the View Patient File method for the Manager | 0.5 | 24 | Programmer C, ScrumMaster |
| 26 | Create graphical user interface of the View Patient File method for the Doctor and Nurse | 0.5 | 25 | Programmer C, ScrumMaster |
| 27 | Create graphical user interface of the View Patient File method for the Receptionist | 0.5 | 26 | Programmer C, ScrumMaster |
| 28 | Connect 'View File' method to each user's View File graphical interface | 0.5 | 25,26,27 | Programmer C |
| 29 | Test/fix bugs found in the import method | 2 | 28 | Programmer C, Product Owner |
| 30 | Integrate and test the methods inside the current build of the system | 1 | 29 | Programmer C |
| **IV** | **Book Appointments for Receptionists** |  |  |  |
| 31 | Create 'Add Appointment' method that adds a booking on the Appointment database | 3 | 23 | Programmer A |
| 32 | Create graphical interface for the Receptionist to book an appointment for the patient | 1 | 31 | Programmer A, ScrumMaster |
| 33 | Connect 'Add Appointment' method to each user's View Patient File graphical interface | 0.5 | 31,32 | Programmer A |
| 34 | Test/fix bugs found in the import method | 2 | 33 | Programmer A |
| 35 | Integrate and final test or fix the methods inside the current build of the system | 1 | 34 | Programmer A |
| **V** | **Cancel Appointments for Receptionist** |  |  |  |
| 36 | Create 'Remove Appointment' method that deletes a booking on the Appointment database | 2 | 28 | Programmer B |
| 37 | Create graphical interface for the Receptionist to cancel an appointment for the Patient | 1 | 36 | Programmer B, ScrumMaster |
| 38 | Connect Remove Appointment method to Cancel Appointment graphical interface | 0.5 | 36,37 | Programmer B |
| 39 | Test/fix bugs found in the import method | 2 | 38 | Programmer B |
| 40 | Integrate and final test or fix the methods inside the current build of the system | 1 | 39 | Programmer A,B, C, Product Owner, ScrumMaster, Client |
| 41 | Sprint review with entire team and client | 0.5 | 29,34,40 |  |
|  | **Sprint 2** |  |  |  |
| 42 | Sprint planning, design and analysis | 0.5 | 41 | Programmer A,B, C, Product Owner, ScrumMaster |
| **VI** | **Update Patient File for Doctor, Nurse, and Receptionist** |  |  |  |
| 43 | Create 'Update Patient' method that updates data on the Patient database | 1 | 42 | Programmer A |
| 44 | Create graphical interface for Update Patient for Doctor and Nurse | 1 | 42 | Programmer A, ScrumMaster |
| 45 | Create user interface for Update Patient for Receptionist | 1 | 44 | Programmer A, ScrumMaster |
| 46 | Connect 'Update Patient' method to update Patient graphical interface for each user | 0.5 | 44,45 | Programmer A, ScrumMaster |
| 47 | Test method and fix bugs found | 2 | 46 | Programmer A, Product Owner |
| 48 | Integrate Update patient file method inside the current build of the system | 1 | 47 | Programmer A |
| **VII** | **Create Staff Schedule for Manager** |  |  |  |
| 49 | Create 'Create Schedule' method that updates data on the Patient database | 1 | 42 | Programmer B |
| 50 | Create graphical interface of Create Schedule method for Manager | 1 | 49 | Programmer B, ScrumMaster |
| 51 | Connect 'Create Schedule' method to Create Schedule graphical interface for Manager | 0.5 | 49,50 | Programmer B |
| 52 | Test method and fix bugs found | 2 | 51 | Programmer B, Product Owner |
| 53 | Integrate Update patient file method inside the current build of the system | 1 | 52 | Programmer B |
| **VIII** | **Hours Worked Reports for Manager** |  |  |  |
| 54 | Create 'Hours Worked Reports' method that retrieves data from the Staff Schedule database | 1 | 42 | Programmer C |
| 55 | Create graphical interface to view the reports of Staff Hours Worked | 1 | 54 | Programmer C, ScrumMaster |
| 56 | Connect 'Hours Worked Reports' method to Staff Hours Worked graphical interface | 0.5 | 54,55 | Programmer C |
| 57 | Test method and fix bugs found | 2 | 56 | Programmer C, Product Owner |
| 58 | Integrate Reports method inside the current build of the system | 1 | 57 | Programmer C |
| **IX** | **Add New Staff for Manager** |  |  |  |
| 59 | Create 'Add Staff' method that updates data on the Staff database | 1 | 48 | Programmer A |
| 60 | Create graphical interface to add new staff members for the Manager | 1 | 59 | Programmer A, ScrumMaster |
| 61 | Connect 'add Staff' method to Add Staff graphical interface | 0.5 | 59,60 | Programmer A |
| 62 | Test method and fix bugs found | 2 | 61 | Programmer A, Product Owner |
| 63 | Integrate Add Staff method inside the current build of the system | 1 | 62 | Programmer A |
| **X** | **Check-In for Receptionist** |  |  |  |
| 64 | Create 'Check-in' method that changes a status of a patient to checked-in | 3 | 53 | Programmer B |
| 65 | Create graphical interface for the receptionist to check in patients | 1 | 65 | Programmer B, ScrumMaster |
| 66 | Connect 'check-in' method to check in graphical interface | 0.5 | 64, 65 | Programmer B |
| 67 | Test method and fix bugs found | 2 | 66 | Programmer B, Product Owner |
| 68 | Integrate and final test or fix the methods inside the current build of the system | 1 | 67 | Programmer B |
| **XI** | **Check-Out a patient** |  |  |  |
| 69 | Create 'check-out' method that changes the status of a patient to checked out | 2 | 58 | Programmer C |
| 70 | Create graphical interface for the Receptionist to check out Patients |  | 69 | Programmer C, ScrumMaster |
| 71 | Connect 'check-out' method to checkout receptionist graphical interface | 0.5 | 69, 70 | Programmer C |
| 72 | Test method and fix bugs found | 2 | 71 | Programmer C, Product Owner |
| 73 | Integrate and final test or fix the methods inside the current build of the system | 1 | 72 | Programmer C |
| 74 | Sprint review with entire team and client | 1 | 63,68,73 | Programmer A,B, C, Product Owner, ScrumMaster, Client |
|  | **Sprint 3** |  |  |  |
| 75 | Sprint planning, design and analysis | 0.5 | 74 | Programmer A,B, C, Product Owner, ScrumMaster |
| **XII** | **Add to queue feature for receptionist** |  |  |  |
| 76 | Create 'add to queue' method that moves Patient on the Queue database | 3 | 75 | Programmer A |
| 77 | Create graphical interface in which Receptionist can view queue an add Patients to it | 1 | 75 | Programmer A, ScrumMaster |
| 78 | Connect 'add to queue' method to the graphical interface | 0.5 | 76, 77 | Programmer A |
| 79 | Test/fix bugs found in the import method | 2 | 78 | Programmer A, Product Owner |
| 80 | Integrate and final test or fix the methods inside the current build of the system | 1 | 79 | Programmer A |
| **XIII** | **Remove from queue for receptionist** |  |  |  |
| 81 | Create 'Remove from Queue' method that removes patient on the Queue database | 3 | 75 | Programmer B |
| 82 | Create graphical interface for the receptionist to remove patients from queue list | 1 | 81 | Programmer B, ScrumMaster |
| 83 | Connect 'Remove from Queue' to queue list graphical interface | 0.5 | 81, 82 | Programmer B |
| 84 | Test method and fix bugs found | 2 | 83 | Programmer B, Product Owner |
| 85 | Integrate and final test or fix the methods inside the current build of the system | 1 | 84 | Programmer B |
| **XIV** | **Patient No-Show Fee for receptionist** |  |  |  |
| 86 | Create 'No-Show Fee' method that updates data on the Patient File in the Patient database | 1 | 75 | Programmer C |
| 87 | Create graphical interface to add a No-Show Fee to Patient File for the Receptionist | 1 | 86 | Programmer C, ScrumMaster |
| 88 | Connect 'No-Show Fee' method to No-Show Fee graphical interface for receptionist | 0.5 | 87, 87 | Programmer C |
| 89 | Test method and fix bugs found | 2 | 88 | Programmer C, Product Owner |
| 90 | Integrate Patient No-Show fee method inside the current build of the system | 1 | 89 | Programmer C |
| **XV** | **Add a sick note fee to a Patient** |  |  |  |
| 91 | Create 'Add Sick Note' method that adds a sick note to the Patient database and returns a fee to be paid | 1 | 80 | Programmer A |
| 92 | Create graphical interface for 'Add Sick Note' method for a Patient | 1 | 91 | Programmer A, ScrumMaster |
| 93 | Connect 'Add Sick Note' method to check-in Patient graphical interface | 0.5 | 91, 92 | Programmer A |
| 94 | Test method and fix bugs found | 2 | 93 | Programmer A, Product Owner |
| 95 | Integrate and final test or fix the methods inside the current build of the system | 1 | 94 | Programmer A |
| **XVI** | **Generate General Reports for the Manager** |  |  |  |
| 96 | Create 'General Reports' method that retrieves data from all of the databases | 1 | 85 | Programmer B |
| 97 | Create graphical interface to view General Reports for the Manager | 1 | 96 | Programmer B, ScrumMaster |
| 98 | Connect 'General Reports' method to General reports graphical interface | 0.5 | 96, 97 | Programmer B |
| 99 | Test method and fix bugs found | 2 | 98 | Programmer B, Product Owner |
| 100 | Integrate General Reports method inside the current build of the system | 1 | 99 | Programmer B |
| **XVII** | **Sending Emails** |  |  |  |
| 101 | Create 'Send Email' method that can retrieves data from Staff Database | 3 | 90 | Programmer C |
| 102 | Create graphical interface to send emails for staff | 1 | 101 | Programmer C, ScrumMaster |
| 103 | Connect 'Send Email' method to Sending Email graphical interface | 0.5 | 101,102 | Programmer C |
| 104 | Test method and fix bugs found | 2 | 103 | Programmer C, Product Owner |
| 105 | Integrate Send Email method inside the current build of the system | 1 | 104 | Programmer C |
| 106 | Sprint review with entire team and client | 1 | 95,100,105 | Programmer A,B, C, Product Owner, ScrumMaster, Client |
|  | Sprint 4 |  |  |  |
| 107 | Sprint planning, design and analysis | 0.5 | 106 | Programmer A,B, C, Product Owner, ScrumMaster |
| **XVIII** | **View Emails** |  |  |  |
| 108 | Create 'View Email' method that can retrieves data from all of the databases and stores it in the Reports Database | 3 | 107 | Programmer A |
| 109 | Create graphical user interface to view received emails for staff | 1 | 107 | Programmer A, ScrumMaster |
| 110 | Connect 'View Email' method to View Email graphical interface | 0.5 | 108, 109 | Programmer A |
| 111 | Test method and fix bugs found | 2 | 110 | Programmer A, Product Owner |
| 112 | Integrate View Email method inside the current build of the system | 1 | 111 | Programmer A |
| **XIX** | **Login Feature** |  |  |  |
| 113 | Create 'Login' method for staff that retrieves data from Staff database | 1 | 107 | Programmer B |
| 114 | Create graphical user interface of login page for staff | 1 | 113 | Programmer B, ScrumMaster |
| 115 | Connect 'Login' method to View Email graphical interface | 0.5 | 113, 114 | Programmer B |
| 116 | Test method and fix bugs found | 2 | 115 | Programmer B, Product Owner |
| 117 | Integrate Login method inside the current build of the system | 1 | 116 | Programmer B |
| **XX** | **Logout Feature** |  |  |  |
| 118 | Create 'Logout' method for staff that retrieves data from Staff database | 1 | 107 | Programmer C |
| 119 | Create graphical user interface of logout page for staff | 1 | 118 | Programmer C, ScrumMaster |
| 120 | Connect 'Logout' method to View Email graphical interface | 0.5 | 107, 119 | Programmer C |
| 121 | Test method and fix bugs found | 2 | 120 | Programmer C, Product Owner |
| 122 | Integrate Logout method inside the current build of the system | 1 | 121 | Programmer C |
| **XXI** | **Scheduled Reports Feature for Manager** |  |  |  |
| 123 | Create 'schedule reports' method that sends any report to an email at any schedule or time interval gets information from Reports Database | 3 | 112 | Programmer A |
| 124 | Create graphical user interface to add schedule or time intervals on sending reports for Manager | 1 | 123 | Programmer A, ScrumMaster |
| 125 | Connect the 'Scheduling Reports' method to graphical interface | 0.5 | 123,124 | Programmer A |
| 126 | Test method and fix bugs found | 2 | 125 | Programmer A, Product Owner |
| 127 | Integrate 'schedule reports' method inside the current build of the system | 1 | 126 | Programmer A |
| **XXII** | **Easy referral feature** |  |  |  |
| 128 | Create 'Send Referral' method for doctor that adds patient file to an email to be sent to another clinic/specialist | 3 | 117 | Programmer B |
| 129 | Create graphical interface for 'Send Referral' method. | 2 | 128 | Programmer B, ScrumMaster |
| 130 | Connect 'Send Referral' method to send referral graphical interface | 0.5 | 117,129 | Programmer B |
| 131 | Test method and fix bugs found | 2 | 130 | Programmer B, Product Owner |
| 132 | Integrate referral method inside the current build of the system | 1 | 131 | Programmer B |
| **XXIII** | **Que Status to "with the doctor" for the nurse** |  |  |  |
| 133 | Create 'With the Doctor' method that shows a patient's queue that he/she is in with the doctor | 3 | 122 | Programmer C |
| 134 | Create graphical interface for 'With the Doctor' method. | 2 | 133 | Programmer C, ScrumMaster |
| 135 | Connect 'With the Doctor' method to with the doctor graphical interface and patient Queue database | 1 | 133,134 | Programmer C |
| 136 | Test method and fix bugs found | 2 | 135 | Programmer C, Product Owner |
| 137 | Integrate referral method inside the current build of the system | 1 | 136 | Programmer C |
| 138 | Sprint review with entire team and client | 1 | 127,132,138 | Programmer A,B, C, Product Owner, ScrumMaster, Client |

1. **Implementation Plan**

**Project Start Sample : January 2, 2019**

