**ПРОЕКТ ПО ПРЕДМЕТОТ ВИЗУЕЛНО ПРОГРАМИРАЊЕ**

**One Minute Sniper**

Иво Тасевски 183132

Матеј Цветановски 183210

Ние, за овој проект се одлучивме да изработиме едноставна игра која воедно ќе биде и интересна за играње и ќе вклучува компетентен дух помеѓу играчите. Името на оваа игра е One Minute Sniper. Како што кажува името, една сесија од оваа игра има временски период од една минута, во која корисникот мора со помош на глувчето и десниот клик да погоди што е можно повеќе балончиња. Корисникот, преку бројни опции, има можност да си ја прилагоди околината на играње со тоа што може да го менува стилот на нишанот, стилот на балончињата и сл. Откако ќе заврши играта, на корисникот му се прикажува дијалог кој го информира дали неговиот high score е надминат и доколку е, истиот се впишува во текст-базирана податочна база која, за секој корисник што ќе се пријави, содржи кориснички информации за неговото корисничко име и неговиот high score, овозможувајќи притоа памтење на секој нов high score за секој корисник, соодветно.

Оваа наша апликација содржи бројни класи и форми, како што се Player класата, која ги содржи претходно споменатите податоци за секој корисник, потоа Reticle класата, која го дефинира стилот на нишанот и Target класата, која го дефинира стилот на балончињата. Поради тоа што има поголем број на форми, како и наведените класи, ние изработивме еден Control Flow Diagram за да прикажеме како сите форми и класи се поврзани и работат заедно на синхронизиран начин. На следната слика е прикажан CFG графот (Напомена: квадратчињата на графот се формите, кругчињата се класите, а цилиндарот е текст-базираната податочна база):

A close up of text on a white background

Description automatically generated

Слика 1. CFG за играта

Како што може да се види според графот, сите форми и класи работат координирано и се поврзани меѓусебно. Кога во Settings формата ќе се зачуваат прилагодувањата, се праќаат преку Main Menu формата, до Gamertag Prompt формата, па од таму се овозможува нивно вметнување во New Game формата. Пред да започне играта, се проверува базата за вметнатиот gamertag со повик до функцијата checkDatabase, која пак прави повик до функцијата readFromDatabase и враќа булова вредност во зависност од тоа дали за тоа корисничко име има запис во базата. Ако има hit, се земаат податоците за корисникот (неговиот gamertag и high score). Во спротивно, се креира нов запис за внесеното корисничко име со помош на функцијата writeToDatabase. Потоа започнува играта. Кога ќе помине времето, се проверува дали high score-от на корисникот е надминат. Доколку да, се прави повик до функцијата writeToDatabase во Gamertag Prompt формата, со што новиот high score е запишан во базата и откако е запишан, се пристапува End Game Dialog формата. Во спротивно, само се пристапува End Game Dialog формата и се прикажува score-от на корисникот, како и неговиот high score и му се дава опција да започне нова игра или да се врати назад на почетниот екран (Main Menu). На овој начин вие сега имате една слика за тоа како работи оваа апликација.

Тука ќе направиме сега и една анализа за претходно споменатите функции кои воедно се и три од најбитните функции на оваа игра:

bool checkDatabase(String player\_gt) : оваа функција прави повик до readFromDatabase функцијата и враќа булова вредност во зависност од тоа дали има запис во базата за player\_gt.

bool readFromDatabase(String gamertag) : оваа функција исто така враќа булова вредност. Во оваа функција, се креира инстанца од класата TextReader од тип StreamReader која ќе чита од базата (PlayerDatabase.txt). Притоа, ќе се изминува секоја линија од текстуалната датотека и ќе се проверува секоја линија дали го содржи вметнатиот gamertag од Gamertag Prompt формата (пратен овде како аргумент). Доколку го содржи, се поставува еден едноставен flag и во зависност од тој flag се праќа булова вредност true или false. Ако flag-от е false, тоа значи дека тој gamertag не постои во базата и се праќа false, а ако е true тоа значи дека постои. Ако е true, се прави split на линијата од датотеката по празно место (Напомена: во класата Player е преоптоварен ToString() методот, и податоците во базата се запишуваат како објекти од класата Player во формат “gamertag highscore”) и двата податока се ставаат во низа со големина од две ќелии кои потоа се сместуваат во соодветните променливи кои ќе бидат пристапени од New Game формата.

void writeToDatabase(Player p) : оваа функција е прилично едноставна и се користи за да се направи запис на објектот p од класата Player во базата. Се креира инстанца од класата TextWriter од тип StreamWriter која ќе запишува во датотеката/базата (PlayerDatabase.txt).

Други две функции кои исто така се многу битни од аспект на играње на самата игра се функциите за настаните MouseClick() и MouseMove() заедно со Paint() настанот и неговата функција:

void NewGame\_Paint(object sender, PaintEventArgs e) : оваа е најкруциелната функција од аспект на секоја игра или апликација која се базира на одредени shapes и нивно цртање и движење, како што е нашата игра. Оваа функција го поставува почетниот нишан да биде во средина и од таа точка корисникот делегира што ќе се случува. Дали со стискање на копчето S ќе започне играта, со P ќе се паузира или со B ќе се прекине играта. Се она што прави корисникот од овде, се праќа на оваа функција која пак диктира што е она што се прикажува на екран, преку клучниот збор Invalidate. Како што се движи глувчето, постојано се извршува void NewGame\_MouseMove(object sender, MouseEventArgs e) функцијата која во реално време праќа x и y координати на Paint со користење на Invalidate, со тоа овозможувајќи движење на нишанот. При секој клик пак, се извршува void NewGame\_MouseClick(object sender, MouseEventArgs e) функцијата која проверува дали балончето е погодено. Доколку е, се брише од екран и се генерираат нови x и y координати кои се рандом координати за следното појавување за следното балонче и преку Invalidate, истите се праќаат на Paint. Со се тоа кажано, настаните MouseMove и MouseClick може да се разгледувааат како помошници на Paint настанот.

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Слика 3. Gamertag Prompt форма

Слика 2. Main Menu форма

A picture containing bird

Description automatically generated

Слика 4. New Game форма – почетен изглед

A picture containing game, table

Description automatically generated

Слика 5. New Game форма – изглед во акција

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

Слика 7. Settings форма

Слика 6. End Game Dialog форма

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Слика 8. About форма и patch notes

Слика 9. Follow Us форма